FOURIER TRANSFORM INFRARED SPECTROSCOPIC MEASUREMENT OF CARBON MONOXIDE AND NITRIC OXIDE IN SIDESTREAM CIGARETTE SMOKE IN REAL TIME USING A HOLLOW WAVEGUIDE GAS CELL AND NON-IMAGING OPTICS

A Dissertation
Presented to
The Academic Family

By

Bruce T. Thompson

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in School of
Chemistry and Biochemistry

Georgia Institute of Technology
May, 2004

FOURIER TRANSFORM INFRARED SPECTROSCOPIC MEASUREMENT OF CARBON MONOXIDE AND NITRIC OXIDE IN SIDESTREAM CIGARETTE SMOKE IN REAL TIME USING A HOLLOW WAVEGUIDE GAS CELL AND NON-IMAGING OPTICS

Approved by

Dr. Boris Mizaikoff, Advisor

Dr. Thomas Orlando

Dr. Jiri A. Janata

Dr. Marcus Weck

Dr. William D. Hunt

Date Approved __May 6, 2004_____

This is dedicated to my wife, Kathy,
my daughter, Natalie,
and all those who had faith in me.

ACKNOWLEDGEMENT

A college degree is not a sign that one is a finished product but an indication a person is prepared for life. - Rev. E. A. Malloy.

Preparation for life is not journey one makes alone, nor achieves without help. So, it is the same for this accomplishment in my life. Having embarked on my graduate career later in life than most, I have relied on the experience of others more than most. It would not be possible to make this achievement at my point in life unless there was someone who was willing to take a chance with me. Dr. Boris Mizaikoff at Georgia Institute of Technology was kind enough to accept me into his group. His confidence in me and his willingness to work within my unique situation inspired me toward my goal. He has shown me that knowledge and wisdom are not always associated with age and neither are youthful ideals and aspirations. For being my advisor, coach and mentor, thank you.

I have essentially worked two jobs to have reached this point. And both employers have been more than accommodating. Both Georgia Institute of Technology, and in particular my committee, and Brown & Williamson have trusted me to complete my goals amidst the flexibility needed to achieve this degree and continue support the business. I am deeply appreciative of their consideration. There are a couple of people I work with at B&W with that deserve special recognition. All have been my supervisor at one time or another and all continue to be mentors and colleagues. I would like to thank

Drs. Kelley St. Charles, Roy Taylor, and Natalie Takenaka for their assistance and support through the years. And, I hope they will continue to extend the same opportunity to others in the future.

I mentioned I worked two jobs, but in reality I have held three. I dedicated this book to my wife and daughter because they have felt the stress and have been patient for more than six years. I could not have accomplished all that I have without their love and support. And their support goes beyond the emotional support everyone needs from their loved ones. Each has accomplishments in the field of chemistry, so my research is not boring. Our discussions at dinner or elsewhere have augmented my discussions at school or work. I cannot express the depth of my appreciation for their love, understanding, and support for me as I pursued my degree.

Finally, I would like to extend a special thank you to a person who helped me start my graduate career. I started my graduate studies at the University of Louisville under Dr. Cecilia Yappert, at a time when part-time Ph.D. students were a very rare occurrence. I thank Dr. Yappert for her willingness and guidance in my academic career.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Δ	Change
Δ_{max}	Maximum mirror displacement
ν	Frequency
θ	Half angle of CPC geometric field of view
μ	Reduce mass
λ	Wavelength
π	Pi
δ	Mirror retardation
$\bar{\nu}$	Frequency in wavenumbers (cm^{-1})
$\nu_{absorption}$	Frequency of absorption
ν_{osc}	Frequency of vibration
ν_{rot}	Frequency of rotation
a	Half size of CPC entrance aperture
a'	Half size of CPC exit aperture
$a_{\bar{\nu}}$	Absorptivity coefficient at frequency $\bar{\nu}$
$A_{\bar{\nu}}$	Absorbance at frequency $\bar{\nu}$
b	Cell pathlength

B	Rotational constant
$B(\bar{\nu})$	Spectral intensity at frequency $\bar{\nu}$
CLS	Classical least squares
cm	Centimeter
cm^{-1}	Wavenumber (recipicol centimenter)
CO	Carbon monoxide
CO_2	Carbon dioxide
CO_x	Carbon oxides
CPC	Compound parabolic concentrator
D	Direction vector for a ray
E_J	Rotational energy level
EPA	Environmental Protection Agency
E_v	Vibrational mode energy level
f	Focal length
FT-IR	Fourier transform infrared
h	Planck's constant
H_2O	Water
HW	Hollow waveguide
I	Incoming ray vector
$I(\delta)$	Interferogram intensity for mirror retardation δ
i.d.	Internal or inside diameter
IR	Infrared

J	Rotational angular momentum quantum number
k	Force constant
L	Length of CPC
M	Intersection point on mirror surface
m	Meter
mg	Milligram
mL	Milliliter
mm	Millimeter
N	Surface normal
NDIR	Non-dispersive infrared
NIOSH	National Institute for Occupational Safety and Health
NO	Nitric oxide
NO_x	Nitrogen oxides
OP/FT-IR	Open path, Fourier transform infrared spectrometry
OSHA	Occupational Safety and Health Administration
P	Point on a ray
P	Power of the transmitted radiation
P_0	Power of the incident radiation
PAH	Polycyclic aromatic hydrocarbon
PCR	Principal components regression
PLS	Partial least squares
q	Change in intermolecular distance

QCL	Quantum cascade laser
r	Radius
R	Reflected ray vector
RMS	Root mean square
RMSECV	Root mean square error of cross validation
RMSEP	Root mean square error of prediction
S	Starting point for a ray
SO_x	Sulfur oxides
t	Time
TPM	Total particulate matter
U	Potential energy
U.S.	United States
v	Vibrational quantum number
VOC	Volatile organic compounds
z	Position in z direction

SUMMARY

The application of a hollow waveguide (HW) was investigated as a gas cell for analytical infrared analysis. The analysis was the measurement of carbon monoxide (CO) and nitric oxide (NO) in sidestream cigarette smoke. An FT-IR analysis system was setup with a 3m multi-pass gas cell and a 55cm by 2mm i.d. Ag/AgI coated HW in tandem with individual CO and NO gas analyzers. The HW demonstrated response times an order of magnitude less than the larger volume multi-pass gas cell and slightly faster than the single analyte gas analyzer. Furthermore, it has been demonstrated that the HW provides up to approx. 60% greater sensitivity on a per meter optical path basis than the multi-pass gas cell of the analytes investigated due to increased optical efficiency maximizing the light concentration within the gaseous sample volume. Simulations in 3-D showed the sensitivity could theoretically improve by more than an order of magnitude if the IR beam was coupled more efficiently into the waveguide. Both FT-IR configurations gave statistically equivalent results for CO to the independent analyzers. With the HW increased temporal resolution, inter-puff measurements comparable to the gas analyzer were achieved at a lower spectral resolution.

The HW optical configuration was modeled for ray tracing in MATLAB. Simulations in 2-D and 3-D were accomplished. The simulations show a major drawback to HW optimization is the coupling of the infrared beam into the waveguide. As demonstrated in a 3-D simulation, approximately 97% of the rays are rejected when an off-axis parabolic mirror with 25.4mm focal length is used to focus the IR beam into the 2mm i.d.

waveguide. Repeating the simulation with longer focal length mirrors showed improved in IR coupling into the waveguide from 3% to 85%. Simulations applying a compound parabolic concentrator show comparable performance to the traditional design of two OAP mirrors to collect rays from the HW and focus onto the detector, but in a much smaller configuration. The simulation routines can be used to further improve the design of this and other optical sensing systems and enhanced by incorporating a spectral component to the simulation.

CHAPTER 1.

INTRODUCTION

In industrial settings, the goal is often can more be done faster and better and cost less. This paradigm has promoted many improvements in the industrial analytical lab. Application of new technologies in the industrial lab requires demonstration of comparable performance to an accepted approach. In the cigarette industry, regulatory requirements push manufactures to report more and more data related to the product. To avoid exponential expansion of the analytical costs, more multi-component analyses are performed. This is easily envisioned with chromatography techniques, but some components of smoke are not amenable to chromatography, such as CO and NO. Currently, small gaseous compounds are measured by individual analyzers. By using FT-IR spectroscopy and chemometrics, these compounds and others can be measured simultaneously.

This thesis presents the application of a new technology to an industrial application, real time infrared measurement of CO and NO in sidestream cigarette smoke. The term real time refers to collection of the data during the experiment as opposed to trapping the analyte and performing the analysis post experiment. The progression of experiments includes demonstrating the ability of FT-IR spectroscopy to measure CO and NO in sidestream cigarette smoke, characterize the HW for measuring the analytes, then apply the HW to the real world application.

1.1. Cigarette combustion processes and smoke generation

Cigarette smoke is the result of the pyrolysis and combustion of tobacco coalescing into an aerosol. The smoke is formed in two distinctive zones[1] (Figure 1–1). The combustion zone takes place in the region of the lit cigarette where temperatures generally exceed 600°C, called the coal. From temperatures that can exceed 900°C at the center of the coal during the puff, the temperature decreases about 100°C per millimeter. This produces a second zone immediately behind the coal where temperatures are below 600°C. In the second zone tobacco is distilled and/or pyrolyzed to produce a more chemically complex mixture of gases and vapors[2-4].

The generation of smoke is the result of a self-sustaining combustion cycle. Tobacco is heated by the exothermic combustion at the coal. The heated tobacco first evolves volatile and semi-volatile compounds, including moisture and nicotine. As the tobacco temperature increases as the coal burns back into the tobacco mass, the tobacco begins to thermally decompose in an oxygen depleted environment. As the coal burns through the pyrolyzed tobacco, the combustion process results in formation of CO, CO_2 and H_2O. The remaining ash is composed of inorganic material. The heat generated from the exothermic combustion transfers to the tobacco, completing the cycle.

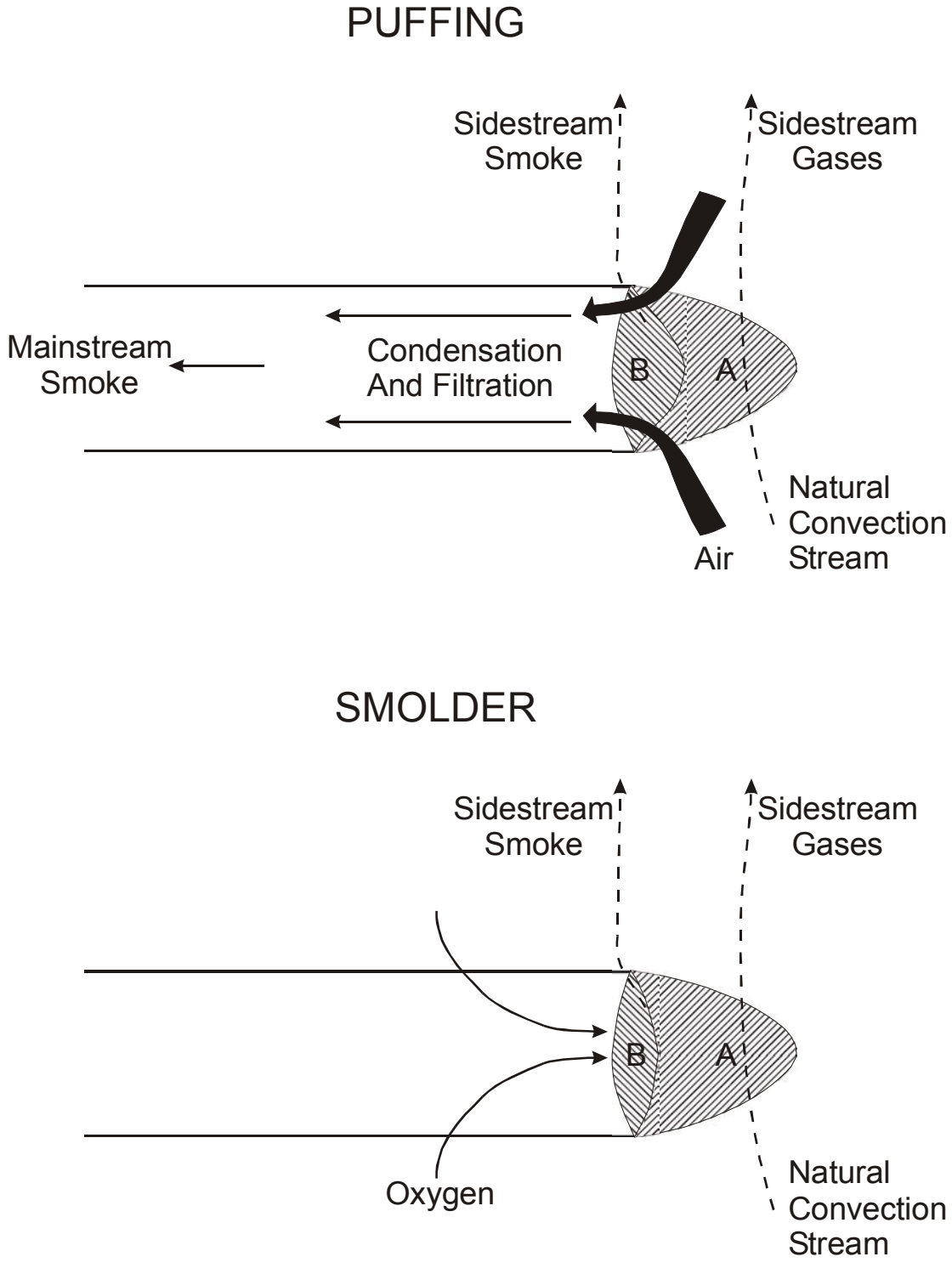

Figure 1–1. Schematic of cigarette combustion during puffing and smoldering. Smoke is generated in two zones: (A) combustion and (B) pyrolysis. (Adapted from Baker[1]).

The combustion cycle results in two burning zones. The combustion zone consumes oxygen by combustion on a carbonaceous substrate. The heat from the exothermic combustion provides the energy for the endothermic processes of the pyrolysis/distillation zone. While the products of combustion are simple gases, the pyrolysis/distillation products are more complex. Examples of classes of compounds that distill from tobacco include aliphatic and terpenoid hydrocarbons, esters and other carbonyl compounds, alcohols, alkaloids and aliphatic amines. The distillates account for about a third of the compounds in cigarette smoke. The efficiency of direct transfer to smoke is dependent on a number of variables, including volatility and thermal stability of the compound. The remaining organic matter consists primarily of biopolymers like polysaccharides (principally cellulose), polyphenols (lignin) and proteins. The biopolymers are subject to thermal degradation resulting in a complex mixture of pyrolysis products, including pyridines, aromatic amines, furans, phenols, and various carbonyl compounds.

The combustion processes take place under two distinctive sets of conditions, puffing and smoldering. During the puff, air is drawn in through the coal increasing the combustion rate and the coal temperature. A typical puff cycle lasts two seconds and can achieve heating rates as high as 500°K s^{-1}. The smoke generated during the puff travels through the tobacco column. The mixture of hot gases and vapors cool as they transfer down the column. Higher molecular weight compounds quickly saturate and condense onto the surface of the tobacco or onto nuclei forming aerosol particles. As the coal burns down the tobacco column, the condensed material on the tobacco either re-

volatilizes or is subject to further pyrolysis or combustion. Also, as the distance from the coal to the filter reduces, the average temperature in the tobacco column increases, permitting more of the higher molecular weight compounds to remain in the vapor or aerosol phase. The result is a more concentrated mainstream smoke further down the column as the tobacco is smoked.

Smoldering is the second type of combustion in the cigarette. During smoldering, combustion is maintained by convective flow around the burning zone. This natural air flow carries the gases and vapors up and away from the cigarette. The combustion kinetics change little during smoldering due to the consistency of air flow around and through the burning zone. Baker[3] showed that little change occurs in the oxygen concentration and temperature profiles of the sidestream cigarette smoke, even during the puff. This demonstrates the independence of the combustion processes occurring at the surface and within the burning zone.

Furthermore, there are two distinctive generation sites for sidestream smoke. In front of the paper burn line where the ash and burning zone are located, the sidestream plume is gaseous. It primarily consists of simple combustion products, like CO, CO_2 and H_2O. The CO concentration just above the surface of the burning zone is slightly higher than inside, indicating carbonaceous reduction of CO_2 to CO on the ash surface. Immediately behind the paper burn line is a denser, visible sidestream plume. The visible sidestream occurs at about the same position as the distillation/pyrolysis zone. The concentrated organic vapor formed in the distillation/pyrolysis zone radially diffuses out of the cigarette through paper made more porous by thermal degradation. Once the vapor has

permeated outside of the cigarette paper, the vapor is subject to both temperature reduction and dilution, favoring the formation of small aerosol particles, which becomes the visible sidestream smoke.

The mainstream and sidestream smoke differ in a couple of important parameters. First is the air flow that generates the combustion. Mainstream combustion is a result of forced air flow during the puff into the combustion zone and tobacco column. Sidestream is the result of natural convections that carry the smoke away from the cigarette. The other distinction is the temperature profiles in the burning zone. Without the forced air flow, there is little change in the heating rates that generate sidestream smoke. Also, the temperature at which sidestream smoke is generated is lower than for mainstream. The resultant relative amounts of pyrolysis and combustion products are different as shown by the range of mainstream to sidestream ratios for various compounds[1,4].

1.2. Gas Phase Infrared Analysis

Absorption of infrared energy is a result of the vibrational motion of molecular bonds. The fundamental infrared absorption is associated with stretching or bending motion of a specific bond or bonds on the molecule under study. Stretching motion results from the movement of the two atomic nuclei towards and away from each other. Diatomic molecules are classic example of stretching motion. During the motion, the distance between the two atoms decreases and then increases from its equilibrium distance. As the internuclear distance decreases, the atoms approach each other and increase the potential energy due to the repulsive nature of their nuclei. As the distance between the

two atoms increases, the potential energy approaches zero as the distance approaches a distance great enough to dissociate the two atoms. Unless additional energy is added to the system, there is insufficient energy to dissociate the atoms and break the bond.

The motion periodically repeats itself and can be modeled following a harmonic oscillator, which is described by

$$\nu_{osc} = \tfrac{1}{2}\pi\left(\frac{k}{\mu}\right)^{1/2}, \tag{1-1}$$

where ν_{osc} is the frequency of vibration, k is the force constant of the hypothetical massless spring connecting the two atoms, and μ is the reduced mass of the two atoms.

The potential energy, U, of the spring is defined by

$$U = \tfrac{1}{2}kq^2, \tag{1-2}$$

where q is the change in internuclear distance from the equilibrium position. Unlike a harmonic oscillator where the potential energy is a continuous function, the introduction of the potential energy function in the Schrödinger equation results in a series of discrete quantum energy levels for the different vibrational modes (E_v) of the molecule, defined by the vibrational quantum number, v,

$$E_v = h\nu_{osc}\left(v + \tfrac{1}{2}\right). \tag{1-3}$$

For example, the fundamental vibrational band is due to the transition from v=0 to v=1. Substituting into Equation (1-3), the energy of this transition is

$$E_{v=1} - E_{v=0} = h\nu_{absorption} = h\nu_{osc}\left[\left(1 + \tfrac{1}{2}\right) - \left(0 + \tfrac{1}{2}\right)\right]. \tag{1-4}$$

Therefore, the fundamental absorption frequency is the oscillation frequency for transition where $\Delta v = +1$.

In addition to vibrational motions, the molecule can undergo rotational motion perpendicular to the bond axis. For linear molecules, the energy associated with rotational transitions is approximated by the rigid rotor model. The rigid rotor model describes the molecule as a dumbbell where the two atoms are connected by a massless rod. The frequency of rotation is defined by

$$\nu_{rot} = 2B(J+1), \tag{1-5}$$

where B is the rotational constant and J is the rotational angular momentum quantum number. Applying the rotating linear system to the Schrödinger equation, the energies of the rotational levels, E_J, are

$$E_J = BJ(J+1). \tag{1-6}$$

Typically, the rotational energy levels are much less that the energy available at room temperature. Therefore, the number of rotational levels appreciably populated could be 30 or more. Transitions between rotational levels require $\Delta J = \pm 1$ for linear molecules and only occur in molecules with permanent dipoles.

Observed infrared spectra of the fundamental vibrational band of small gaseous diatomic molecules, like CO and NO, contain a large number of absorption bands that correspond to the rotational energy transitions. The rotational bands for the fundamental vibration band arise from a combined rotation-vibrational transition. The energy of the rotation-vibrational transition is defined by combining the energy for each transition,

$$E_{v,J} = \nu(v+\tfrac{1}{2}) + B_v J(J+1) \text{ for } v = 0,1,2,\ldots \text{ and } J = 0,1,2,\ldots \tag{1-7}$$

where B_v denotes the dependence of the rotational constant on the vibrational quantum number, v, due to changes in the distance between the atoms while the molecule

oscillates. The selection rules for rotation-vibrational transitions at the fundamental vibrational band are $\Delta v = +1$ and $\Delta J = \pm 1$. Since many different rotational levels can be populated at ambient temperature, many different transitions of differing energies can occur (Figure 1–2). When $\Delta J = +1$, a series of absorptions will be seen in the spectrum at energies greater than the energy of the fundamental vibrational transition. This series of absorptions are designated the R-branch. Likewise, the series of absorptions at energies less than the energy of the fundamental vibrational transition due to $\Delta J = -1$ is designated the P-branch.

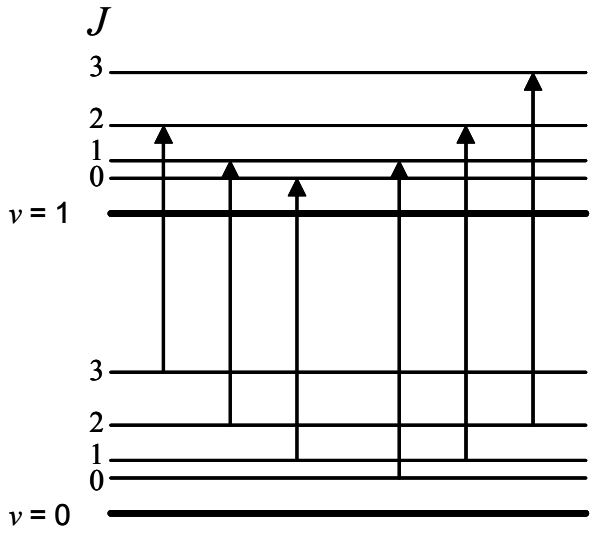

Figure 1–2. Rotational energy transitions at an infrared vibrational fundamental absorption.

There is also a Q-branch that arises when $\Delta J = 0$. However, this is a forbidden transition in linear molecules except when there is an odd number of electrons, as with NO. Rotational bands are evident only in gas phase spectra collected at sufficiently high resolution. These bands are not resolved in condensed phase spectra because of frequent collisions between molecules. The result for condensed phase spectra is broad absorption bands occurring at the vibrational transition energies.

1.3. Quantitative Analysis

The Beer-Lambert law expresses the relationship between sample concentration, *c*, and the amount of infrared radiation absorbed at the measured frequency (in wavenumbers, $\bar{\nu}$), $A_{\bar{\nu}}$,

$$A_{\bar{\nu}} = -\log\left(P/P_0\right) = a_{\bar{\nu}}bc\,, \tag{1-8}$$

where P_0 is the power of the incident radiation, P is the power of the transmitted radiation, $a_{\bar{\nu}}$ is the absorptivity coefficient and is dependent on the frequency of the infrared radiation, and b is the optical pathlength. Although theory indicates a linear relationship, in practice, the linearity of the curve is dependent upon the concentration range and other instrumental factors. Thus, for a given sample matrix, concentration range, and operating conditions, the absorptivity must be determined empirically. There are several calibration techniques used in infrared spectroscopy to predict the sample concentration. These include univariate approaches utilizing absorption peak heights or integrated absorbance, or approaches that utilize more of the spectral information, like classical least squares or partial least squares called multivariate techniques.

1.3.1. Univariate Calibration

Univariate calibrations regress concentration against an absorbance metric, like peak height or integrated absorbance. Peak height is the absorption measurement at a specific frequency, typically the absorption maxima for the fundamental vibrational band. However, depending upon the analyte and the calibration range, shoulders of the absorption band or weaker rotational bands may be used for quantitative purposes. Non-linear relationships between concentration and absorption can be handled with univariate calibrations by performing higher order regressions, such as quadratic regressions.

Peak heights are measured as the difference of the measured absorbance at a specific frequency and the baseline. The baseline can be determined by various methods. The simplest is to arbitrarily set the baseline at zero. For spectra collected under controlled, repeatable conditions, this is a sufficiently accurate approximation. More realistically, additional regions of the spectrum need to be used to determine the baseline. Non-absorbing regions or frequencies on either side of the absorption band provide an improved estimate of the baseline. These regions can be fitted to a straight or quadratic line. The quality of the baseline is critical to the quality of the calibration. The amount of absorption is intrinsic to the quantity and type of molecule. However, the baseline is determined by the measurement conditions during spectra acquisition, including reference spectra and the signal-to-noise ratio of the system.

Integrated absorbance is measure from the spectra by

$$A_{intg} = \int A \, d\bar{\nu} = bc \int a \, d\bar{\nu} \, . \tag{1-9}$$

By integrating over an absorption band, the absorbance integral takes into account all the quantum transition probabilities associated with the absorbance maximum. As derived from Equation (1-9), the integrated absorbance is linearly related to the concentration. Therefore, similar regressions to those used for peak height can be applied to integrated absorbance.

1.3.2. Partial Least Squares

Multivariate calibrations are of increasing importance in the field of analytical chemistry, specifically, quantitative infrared analysis. Partial least squares (PLS) regression is one of several regression algorithms that utilize a wide range of spectral information present in infrared data, hence, often referred to as full spectrum calibrations. The advantages of full spectrum calibrations, like PLS and classical least squares (CLS), are improvements in precision and robustness over univariate methods due to increased signal averaging from including more spectral intensities. The distinction between PLS and CLS substantiates in PLS being a factor based regression, which means the full spectra for the data is reduced to a smaller number of variables that account for the variability in the original data. Furthermore, PLS calibrations can be developed without knowing the entire composition of the sample.

In this respect, PLS is very similar to principle components regression (PCR). Both PLS and PCR are full spectrum methods that decompose the spectral data into two other variables, loadings and scores. The loadings are used to represent the original spectral data. Examination of loadings by a spectroscopist can provide insight into which spectral features are mainly influencing the PLS or PCR score. The scores are reduced versions

of the original spectral intensity data and can be used to formulate linear relationships with the concentration data. PLS and PCR are different in how these linear relationships are established. PCR regresses the concentration vectors for the different components against the scores of the spectral data. This approach does not guarantee that the information contained in a given score relates to concentration. Alternatively, PLS factors the concentration data at the same time as it factors the spectral data. This results in a calibration with better correlation(s) between the reduced spectral data.

The PLS algorithm is shown in block form in Figure 1–3. The spectral data are rows in the spectra data array with the columns representing the intensity at the different frequencies. The columns in the concentration data array are the different components to be quantitated. Each row of the concentration matrix consists of the component concentrations for the corresponding spectral data row.

The PLS algorithm centers and normalizes the raw data. Then the normalized data is reduced to a group of latent variables consisting of scores and loadings. To maximize the covariance between the scores from the concentration and spectra data matrices, PLS iteratively exchanges row data between the scores. The algorithm results in a series of PLS factors that correlate the variability in the spectral data with the variability in the concentration matrix and can be used to predict concentrations in future samples that have a similar composition to the standards used to develop the calibration.

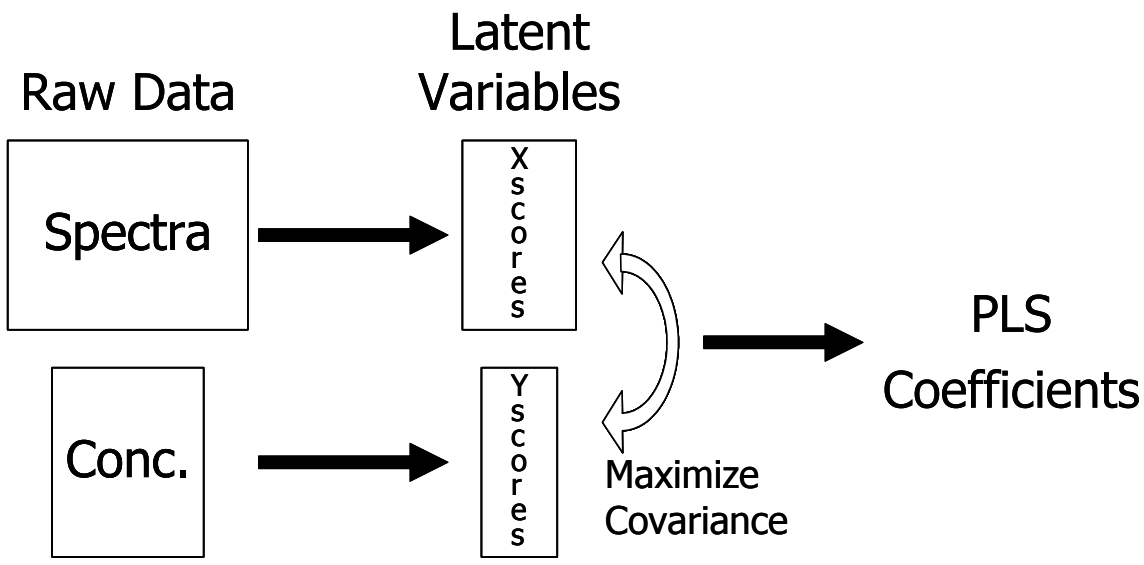

Figure 1–3. Block diagram of PLS algorithm.

Because PLS reduces the dimensionality of the concentration matrix, it removes redundancy and collinearity in the data. Therefore, it is of essential importance to minimize collinearity between components to be measured. Usually, calibrations sets for PLS are uniquely prepared for a given application. One approach is to select a calibration set from a larger empirical data set that represents the expected range and composition of samples to be predicted. Other approaches develop specific calibration sets based on the general guidelines that the range of each component should be uniform and exceed the expected range in the samples, or use more statistical approaches like a factorial design.

Table 1–1. "Minimum correlation table" for 15 standards of a four component mixture.

Standard	Component			
	A	B	C	D
1	1	3	9	11
2	2	13	15	1
3	3	11	6	12
4	4	5	10	4
5	5	8	1	3
6	6	4	13	14
7	7	12	5	15
8	8	15	11	10
9	9	10	4	13
10	10	2	12	6
11	11	7	8	8
12	12	9	14	7
13	13	6	3	5
14	14	14	7	2
15	15	1	2	9

The "minimum correlation table" (Table 1–1) is a general approach to developing PLS calibrations or other calibrations where confounding influences between components are minimized. The "minimum correlation table" is derived from a table of randomized trials[5]. These tables were designed such that correlation is minimal between the orders of the runs during a test with 16 or 32 trials. Applying this table to calibration theory, the range of each component is uniformly divided into the number of levels corresponding to the number of trials. The "minimum correlation table" denotes which level should be used in each standard. If the number standards needed is different than 16 or 32, the table can be modified while monitoring the correlation coefficient between the components.

This novel method of generating a calibration set developed in this dissertation simplifies PLS calibration development, which makes PLS more accessible to routine practitioners.

1.4. Instrumentation for Infrared Gas Analysis

1.4.1. Non-Dispersive Infrared (NDIR) Analyzer

NDIR analyzers are commonly used for single component analyses. The principle of the analyzer is based on the Beer-Lambert law. However, instead of measuring absorbance from a zero baseline, the measurement is determined from a known level of the analyte. The instrument consists of an infrared source whose beam is chopped and split to pass through two gas cells of identical pathlength (Figure 1–4). The reference gas cell is filled with an inert, non-infrared absorbing gas. The sample flows through the other gas cell. The detector is a gas cell with two chambers where the chambers are divided by a thin metal diaphragm, which is one plate of a capacitor. Both chambers of the detector are filled with the same gas sample containing the analyte, which results in selective absorption of infrared radiation.

If the sample and reference gas cells contain the same level of analyte, both chambers of the detector absorb the same amount of infrared radiation and the diaphragm does not deflect. If the sample gas contains the analyte, the detector chamber for the sample beam does not receive as much infrared radiation and therefore is heated less than the reference. This causes the diaphragm to deflect, resulting in a change in capacitance that can be electronically measured.

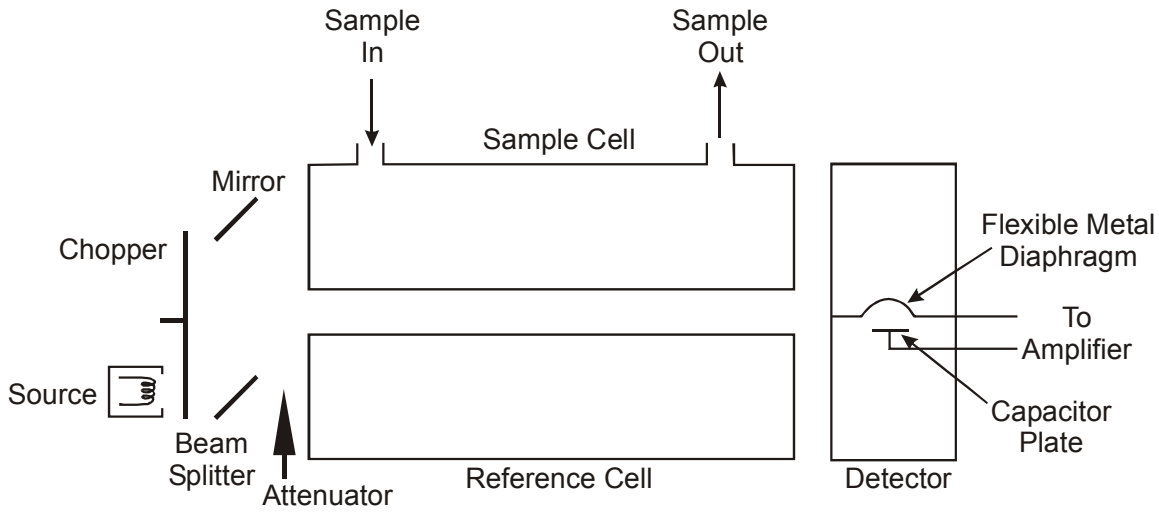

Figure 1–4. Block diagram of a non-dispersive infrared analyzer. The gas cells in the detector are filled with the specific analyte.

The NDIR has a high degree of specificity due to integration of the analyte gas in the detector. The design also limits the instrument to a single gas phase component. To perform multi-component analyses, a unique optical system is needed for each component.

1.4.2. Dispersive Infrared Spectrophotometers

Dispersive infrared spectrometers use optical elements to select a specific frequency of infrared radiation to illuminate the sample. Typical dispersive elements include gratings and narrow bandpass filters. The selected frequency is actually a range of frequencies due to the limitations of the monochromator. Both instruments perform quantitative gas analysis based on the Beer-Lambert law. The broadband source is reduced to the selected frequency by the monochromator. The resultant infrared beam passes through a gas cell. If gas phase molecules absorbing at the selected infrared frequency are present, the

amount of energy reaching the detector is attenuated. By measuring the amount of transmission with and without the analyte of interest present, absorption measurements are obtained.

Dispersive instruments do not have the degree of specificity of the NDIR because the incident radiation covers a small range of frequencies defined by the instruments bandpass. Other components in the sample matrix may absorb at the same frequencies, unless the selected frequency is truly unique for the analyte of interest. Because of the potential for interferences when using a dispersive instrument, measurements at multiple frequencies are collected and multivariate calibrations are typically used.

1.4.3. Fourier Transform Infrared (FT-IR) Spectrometers

Fourier transform infrared spectroscopy is one of several non-dispersive optical spectroscopies based on interferometry. The two beam interferometer first proposed by Michelson is the basis of most modern FT-IRs, as exemplified by the schematic of the Bruker Equinox 55 (Figure 1–5) applied during the studies in this dissertation. Simply described, the interferometer is composed of a beam splitter and two mirrors. A collimated beam of infrared radiation is divided into equal halves at the beam splitter. 50% of the energy travels through the beam splitter to a fixed mirror, which is located at a defined distance away from the beam splitter. The reflected beam travels perpendicular to the incident beam to a moving mirror. The infrared beams reflect off the fixed and moving mirrors and recombine at the beam splitter. The recombined infrared beam projects from the interferometer toward the detector on a path perpendicular to the source beam.

If the two mirrors are equidistant from the beam splitter, the reflected infrared beams are in phase and constructively recombine to yield the maximum energy exiting the interferometer. The mirror positions where the constructive interference is at a maximum is the point of zero path difference. If we consider a monochromatic source of wavelength, λ, constructive interference occurs when the optical path difference (retardation) is a multiple of λ.

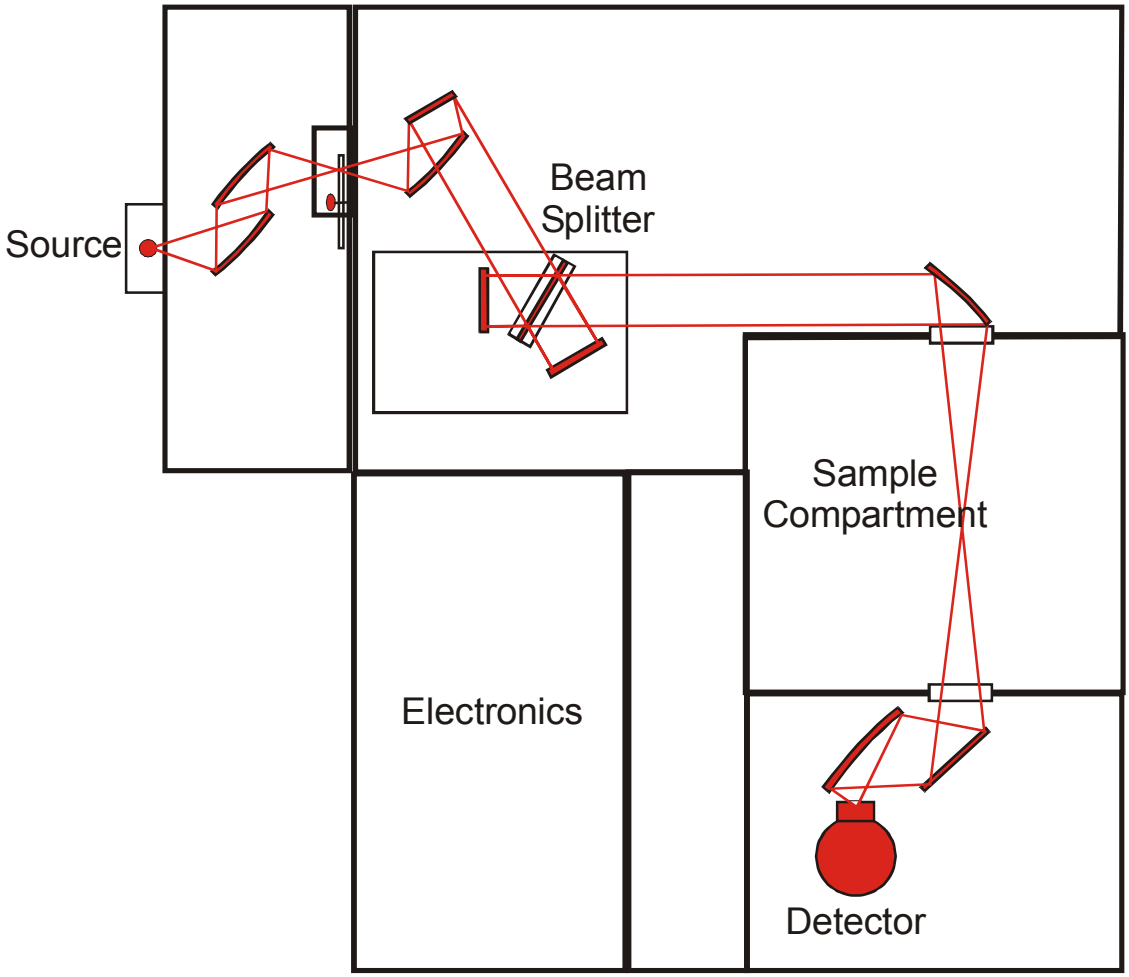

Figure 1–5. Block diagram of Bruker Equinox 55 FT-IR.

If the retardation is λ/2, as when the difference in the distance between the beam splitter and the two mirrors is λ/4, the reflected beams are 90° out of phase and destructively interfere at the beam splitter yielding no energy. By adjusting the distance of the moving mirror around zero path difference, a double sided interferogram is constructed by measuring the infrared energy at the detector. The periodic wave for monochromatic light is expressed as a cosine function,

$$I(\delta) = B(\bar{\nu})\cos(2\pi\bar{\nu}\delta), \tag{1-10}$$

where $I(\delta)$ is the interferogram intensity as a function of retardation, δ, and $B(\bar{\nu})$ is the source intensity as a function of frequency, which includes the instrumental characteristics. If the source is a continuum (broadband) emitter instead of a monochromatic light source, the interferogram is the summation of the cosine functions for all the frequencies present in the beam,

$$I(\delta) = \int_{-\infty}^{+\infty} B(\bar{\nu})\cos(2\pi\bar{\nu}\delta)\cdot d\bar{\nu}, \tag{1-11}$$

which is one-half of a cosine Fourier transform pair. The other half of the pair is the spectrum as a function of frequency and is obtained by computing the Fourier transform of Equation (1-11),

$$B(\bar{\nu}) = \int_{-\infty}^{+\infty} I(\delta)\cos(2\pi\bar{\nu}\delta)\cdot d\delta. \tag{1-12}$$

The maximum resolution is restricted by the maximum retardation of moving mirror in the interferometer. In order to resolve two spectral lines of unique frequencies $\bar{\nu}_1$ and $\bar{\nu}_2$, the interferogram must include when their cosines are both in and out of phase, which

corresponds to a retardation of $\delta = (\bar{\nu}_1 - \bar{\nu}_2)^{-1} = (\Delta\bar{\nu})^{-1}$. If the interferometer has a maximum retardation of Δ_{max}, then the maximum resolution is $\Delta\bar{\nu} = (\Delta_{max})^{-1}$. Consequently, to increase the spectral resolution, the moving mirror must travel farther to achieve increased retardation.

The rate at which data are collected with an interferometer is dependent on how far the moving mirror must travel, *i.e.* the resolution, and how fast the mirror travels, *i.e.* the scanner velocity. Modern FT-IR spectrometers are based on interferometers with scanning velocities able to collect tens of spectra per second at 8cm^{-1} spectral resolution. With faster data collection capabilities, FT-IR spectroscopy can be used to directly observe dynamic and kinetic processes. To monitor such a process, data is collected in the time resolved spectroscopy (TRS) mode. The TRS mode repeatedly collects interferograms according to the acquisition parameters and intermittently stores them in the memory. The interferograms are processed at the end of the data acquisition period. The result of the experiment is a three dimensional data block, where each slice in the block is the spectrum for a time slice in the experiment equal to the interferogram acquisition time.

1.5. Infrared Gas Analysis Applications

Infrared spectroscopy is applied to many types of environmental analyses. These include the analysis of waste streams, soils, and water. Because of the ease in sample handling, infrared analysis of gases, vapors, aerosols, smoke and exhausts in air is the most common application of this technique[6-10]. The types of compounds under surveillance range from small diatomic molecules, like CO and NO, to complex mixtures

of gases and aerosols, like combustion of organics or petroleum fuels[11,12]. Likewise, the purpose of surveillance can vary from threshold monitoring to alert on dangerous condition, such as plume detection, to mapping volatile organic compounds (VOCs) emitted from farms and smokestacks by remote sensing[13,14].

Air quality is important from both a health and safety perspective. In the United States (U.S.), the National Institute for Occupational Safety and Health (NIOSH) and the Occupational Safety and Health Administration (OSHA) set limits of exposure to over 2000 different chemicals or classes of respiratory irritants[15,16]. However, potential exposure is not limited to industrial or laboratory environments. Household and small business environments are equally susceptible to exceeding exposure limits. Many households in the U.S. use natural gas for heating or cooking. A major by-product of natural gas combustion is CO, which needs to be properly exhausted. Also, cleaning products and paints may release relatively high concentrations of VOCs.

It is unlikely that common households will utilize IR gas analyzers monitoring their environments. However, non-scanning IR gas analyzers, like NDIR, can be configured into portable systems applicable to monitoring indoor air quality problems in homes or office buildings. These analyzers are typically much lower in cost than scanning interferometers and can be targeted to detect the presence of one or two specific compounds, like CO and NO. Since quantitative infrared absorption spectroscopy is a function of the pathlength of the gas cell, both non-scanning and scanning instruments can be configured to measure targeted compounds from the low ppm (v/v) range to

percent level concentrations. Measurements at the lower concentrations are obtained by using long pathlength or multi-pass gas cells (see Section 1.6.1).

Many of the compounds monitored indoors are of equal interest in the atmosphere and stratosphere. VOCs are continuously monitored from industrial operations as required by US and local Environmental Protection Agencies (EPA). Power plants and waste incinerators are required to follow emission guidelines for harmful combustion gases, including CO_x and NO_x, as well as other combustion by-products, like polycyclic aromatic hydrocarbons (PAHs), phenols, and hydrocarbons. Most of the compounds have molecule specific infrared signatures that can be used for qualitative and quantitative determinations.

Essentially, the same instrumental methods used for measuring and monitoring indoor air quality are used in outdoor and emission monitoring. In addition to the instrumental methods discussed earlier, open path infrared (OP/FT-IR) spectrometry is used to determine both exogenic and endogenic gases in the atmosphere. In addition to exogenic industrial gases, endogenic gases naturally emitted from plants and geothermal processes can be investigated. These include metabolites of pheromones, terpenes and creosols, as well as photooxidation products.

Ground based OP/FT-IR is achieved by placing as retro-reflecting mirror at the opposite end of the monitoring path aligned with the FT-IR. Such a configuration can be used for monitoring fugitive emissions or fence line monitoring. OP/FT-IR can also be used from a remote sensing platform. This configuration has the advantages of covering a much broader area than ground based systems and being able to measure both

absorption and emission. Passive IR sensing can be used to investigate physical and chemical processes in the atmosphere, monitor industrial emissions and detect climate changes as a result of human activities, like biomass burning and automobile exhaust. Most notable for the latter group of applications are the greenhouse gases, which are contributing to global climate changes.

The determination of combustion products is the prevalent gas analysis application for infrared spectroscopy. These compounds include the oxide products, CO_x, NO_x, and SO_x, and various hydrocarbons (*i.e.* alkanes, alkenes, PAHs, aldehydes and ketones). Combustion gases are emitted from smoke stacks and chimney flues, automobile and jet exhaust, power plants, biomass burning and cigarettes. In addition to off-line monitoring of these and other sources, on-line direct measurement of combustion products can be accomplished with infrared instrumentation. Both CO and CO_2 are continuously monitored in smoke stacks with NDIR analyzers. FT-IR is being increasingly used to simultaneously monitor many of the oxides and hydrocarbons in combustion applications. Car exhaust monitors are used in many states and cities in the U.S. to test vehicle emission compliance. These instruments simultaneously measure eight to ten target compounds. Monitoring of smoke stacks of power plants is not only useful for emission compliance, but also as a control tool to optimize the power plant efficiency. The major advantage of on-line infrared techniques over extractive techniques is the rapid feedback on the gas composition and concentration variations, usually on the order of seconds.

1.6. Infrared Gas Cells

By definition, any IR transmitting cell is a gas cell. Deliberate selection of the optical pathlength renders gas cells particularly appropriate for the detection and quantitative analysis of the compound(s) under study in this dissertation. As described above, the measurement of an IR absorbing compound at a certain concentration is dependent upon the compounds molar absorptivity, which is a constant for a specific frequency, and the optical cell pathlength, which is a variable of selectable dimensions. For strong IR absorbers or at high concentrations, transmission gas cells with pathlengths on the order of 5 to 20cm are suitable. To obtain measurements at lower concentrations, longer pathlength gas cells are required. This section will describe two types of long path gas cells, multi-pass gas cells ("White cells") and hollow waveguides.

1.6.1. Multi-pass Gas Cell ("White cell")

The multi-pass gas cell, also called the "White" cell, consists of two objective mirrors and a field mirror[17] (Figure 1–6). The pathlength is dependent upon the number of passes and the distance between the objective and field mirrors. The longer the pathlength of the gas cell is, the greater the distance between the mirrors and/or the size of the mirrors and thus the greater the cell volume. Many alternative configurations on White's design have been commercialized. They range from cells with less than 2m pathlength and 100mL volume to 20L for 100m gas cells. The cell is usually placed in the sample compartment of the infrared spectrometer or externally coupled. The transfer optics are specific to each spectrometer with the focal point of the infrared beam delivered by the spectrometer at the field mirror of the gas cell. In the simplest configuration, the IR beam enters

through the gas cell window and propagates toward the first objective mirror. The reflection from the first objective mirror is focused onto and reflected off of the field mirror. The reflected beam then propagates to the second field mirror. The reflection from the second field mirror exits the gas cell and is transferred to the detector of the instrument. Therefore, the pathlength of a multi-pass gas cell is a multiple of four of the distance between the objective and field mirrors.

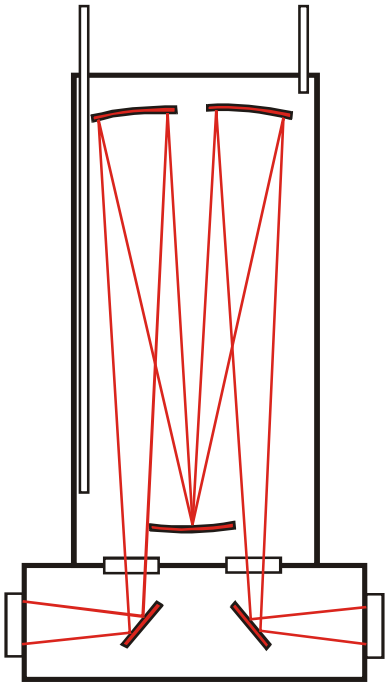

Figure 1–6. Diagram of the multi-pass gas cell with a minimum of four passes of the infrared beam.

The optical efficiency, defined as the ratio of the optical pathlength to the cell volume, can be slightly less than the calculation from the specified pathlength and cell volume of

the gas cell. In the multi-pass gas cell, the reflections produce a series of radiation cones between the objective and field mirrors. Therefore, the IR beam does not pass through the entire volume of the gas cell. For example, the 3m multi-pass gas cell from Gemini Scientific (Buena Park, CA) with a cell volume of 375mL has IR beam coverage of approximately 85% of the 250mL volume defined by the mirrors. Longer pathlength multi-pass gas cells have increased absorption for trace level components due to the Beer-Lambert law. However, because of transmission losses due to multiple reflections on the metal mirrors, the signal-to-noise ratio of the measurement simultaneously decreases.

1.6.2. Hollow Waveguide Gas Cell

An infrared hollow waveguide (HW) is defined as a tube constructed of metal, glass or other dielectric material with the interior surface coated with an IR reflecting material that enables infrared radiation to propagate by reflection off the interior surfaces[18]. Figure 1–7 diagrams in two dimensions how infrared radiation propagation occurs at the entrance and exit of the waveguide. The HW has several advantages as a gas cell over conventional designs. One advantage is the smaller internal volume of the HW cell, which intuitively implies a shorter response time. However, also contributing to the shorter response time is the flow pattern within the cell. The HW, being a tube, establishes a laminar flow through the tube, which is more efficient at transporting gaseous species through the cell. In contrast, the multi-pass gas cell has sample introduction and exhaust port diameters much small than the cell diameter. The difference in diameters, as well as internal structures within the gas cell volume, creates

complex gas flows, including eddies and dead volumes, which increases the time needed to entirely move a gas sample through the cell.

The optical properties of HWs have been extensively studied. The different tube substrates investigated include metal[19], glass[20-22], and plastic[23-25]. The glass (fused silica) and plastic substrates have a flexibility advantage over the metal and crystalline substrates. All HW designs (geometry, substrate, coating and other parameters) show attenuation losses associated with the respective waveguide material due to reflection, scattering and acceptance angle. Even so, HWs have been applied to a variety of infrared sensing applications, including liquid and gas analysis.

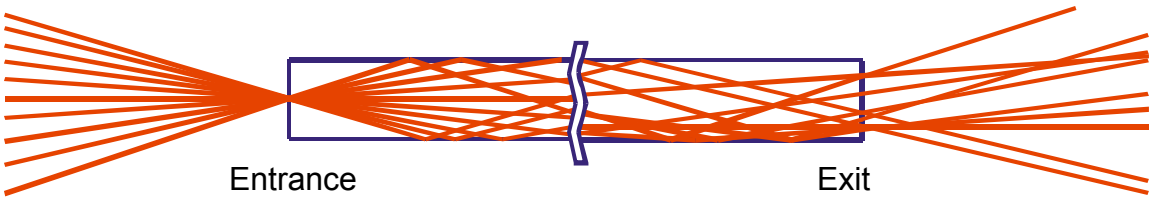

Figure 1–7. Diagram of infrared radiation propagating through a hollow waveguide.

1.7. Monte Carlo Simulation of Optical Systems

Monte Carlo is a mathematical method to numerically determine the behavior of a system[26]. It uses random variables to perturb the system and monitors the results to assess the outcome. The classic example is determining the fraction of the area occupied by the circle of the square that circumscribes the circle. Using random numbers for x and y coordinates of the area in the square and determining whether the point is in the circle or not, an estimate of the circle's area to the squares area can be attained. Performing a million experiments, the estimate can be determined within 0.2% accuracy.

Monte Carlo simulations have been applied to ray tracing problems. The process can be simply described as mathematically defining the surfaces, randomly casting rays and following their fate.

1.7.1. Mathematical Modeling of Optical Components

Optical simulation, specifically ray tracing, involves tracking the path of a ray from its source to the end of the optical system, the detector, or until it is lost or reduced in intensity to a minimum because of absorption or scattering. The optical components in the simulation must be mathematically modeled or defined as a parametric surface in order to calculate ray-surface intersections and surface normals. The normals are used to calculate the direction of a reflection at a specific point on the surface. Four different optical elements are used in the simulation of the HW gas cell. They include the off-axis parabolic (OAP) mirror, the HW, the detector, and the compound parabolic concentrator (CPC). The simulations are developed in 2-D and 3-D environments.

1.7.1.1. Optical Components in 2-D

The OAP mirror is a section from a parabolic surface that does not include the axis of rotation. In 2-D, the surface coordinates, (x,y), are defined by Equation (1-13). The parabola has the y-axis as the axis and the focus is at *f*. Selecting the range of x values defines the segment of this curve to be used as the OAP mirror surface. The position of the OAP mirror surface and the direction of the focus can be changed by multiplying the individual (x,y) coordinates by a rotation/translation matrix. There are three mirrors in the simulation. The first mirror focuses the rays into the entrance of the HW. The second mirror collects the rays emitting from the HW and projects them toward the third mirror, which focuses them onto the detector.

$$y = \frac{x^2}{4f} \tag{1-13}$$

The HW is simulated in 2-D as two horizontal lines the length of the HW and separated vertically by the inside diameter of the HW. Since the long axis of the HW is to be parallel to the x-axis, the HW is positioned by simply translating the coordinates. The final component is the detector. This is a line centered at the focus of the third mirror and extends in both vertical directions ±½ the height of the detector element. The complete simulated optical components are shown in Figure 1–8.

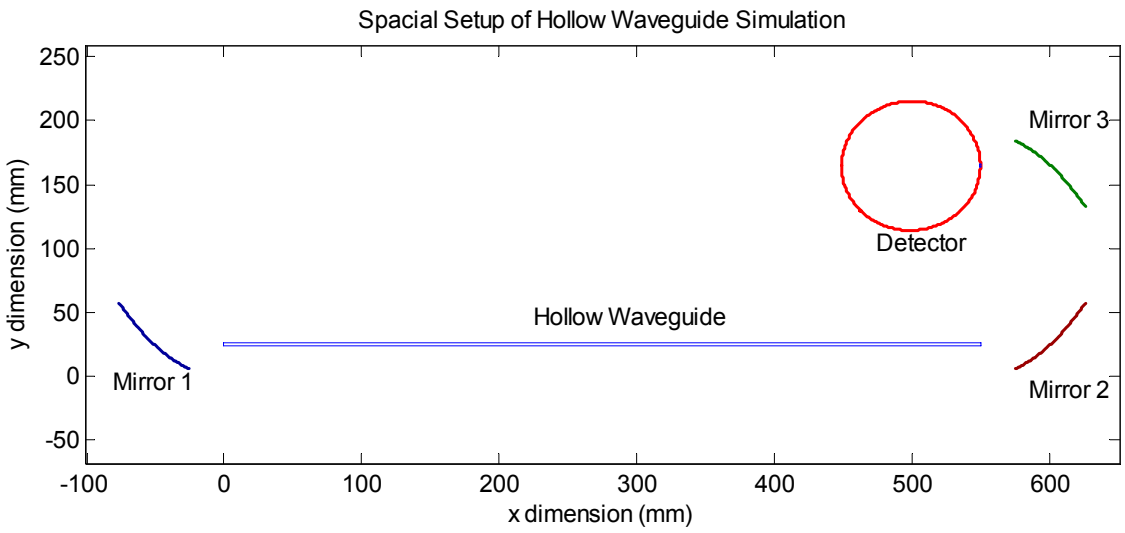

Figure 1–8. Optical layout for 2-D simulation of hollow waveguide.

The final component is the CPC. The upper surface is described by a parabolic curve (see section 1.7.1.3). The curve is reflected across the x-axis to define to lower surface of the CPC. Both curves are translated to a position offset from the end of the HW to model the real positioning due to supporting hardware of the HW. In the cases where the CPC is used, the detector is place immediately after the CPC. Although this concept has been applied to astronomy, it is a novel design concept for infrared sensing applications and can be exploited for more compact instrumentation utilizing waveguides. The detector can be designed with the CPC and the detector element within the Dewar of the liquid nitrogen cooled MCT detector.

1.7.1.2. Optical Components in 3-D

Development of the physical layout in 3-D is an extension of the 2-D layouts. The equation to describe the OAP mirror is identical to Equation (1-13) with the addition of the z-axis component (Equation (1-14)). The section of the paraboloid is defined by a box defined by center coordinates and a length vector for each dimension. To focus rays from the +y direction onto an yz point in the +x direction (mirror 1), the paraboloid has a positive focal point and the box center x point would be negative. Likewise, the box center x point and focal point is positive to focus at an yz point at a negative x position. This mirror configuration is used to collect rays from the HW and project them towards the third mirror in a +y direction. The third mirror is identical to the second mirror except the focus of the paraboloid is negative.

$$y = \frac{x^2 + z^2}{4f} \tag{1-14}$$

The HW is modeled as a cylindrical reflective surface. MATLAB has a *cylinder* function that takes either a scalar for the radius of the cylinder or a vector to describe the geometry of the surface that is parallel to the axis of rotation and the number of segments to parameterize the 2π rotation. The *cylinder* function produces a cylinder of length one. In the case of the HW, this result is replicated to achieve the length of the HW in millimeters. In the case of the CPC, the parabolic curve is provided as a vector and the output from *cylinder* is multiplied by the calculated length of the CPC. The detector is modeled as a square of the desired size for the MCT element. The optical components are translated to specified locations in the layout (Figure 1–9 and Figure 1–10).

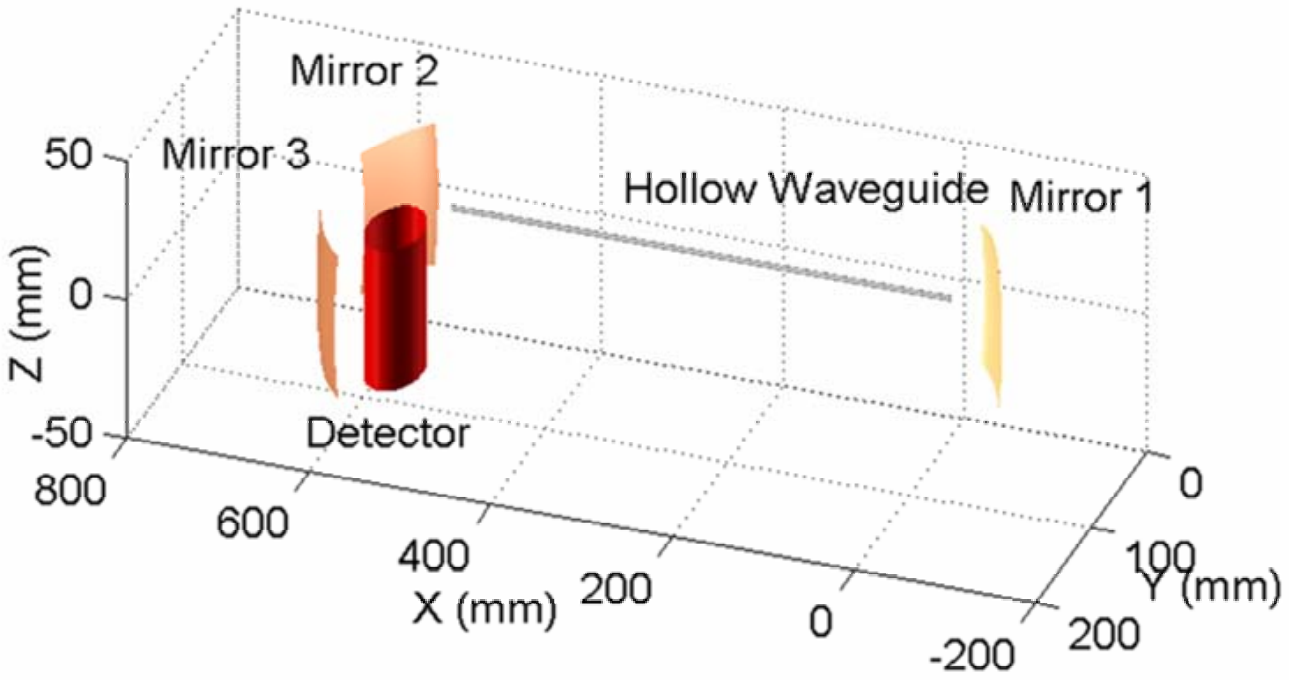

Figure 1–9. Three dimensional layout of hollow waveguide simulation using OAP mirrors.

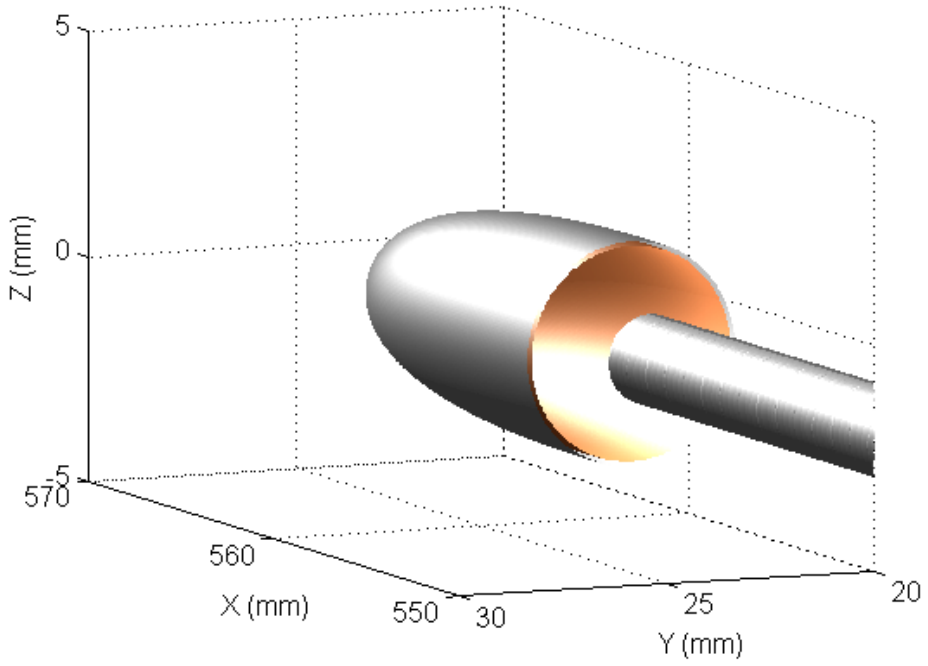

Figure 1–10. Three dimensional rendering of CPC at the exit of the HW (detector not shown).

1.7.1.3. Compound parabolic concentrator (CPC)

The field of non-imaging optics and compound parabolic concentrators was originated in the 1960's[27-30]. In the U.S., research into the CPC was performed by Dr. Roland Winston and the concentrator cone often bears his name, the Winston Cone. The premise of non-imaging optics is collect light that is normally lost by imaging optics due to blurring and broadening of image rays away from the focus. Therefore to achieve maximum collection of all rays, the optical design must forego the imaging requirement.

In the application of the HW, the waveguide itself is a non-imaging optical component, so it is equally applicable to use non-imaging optics to collect the emitted rays from the HW and focus them onto the detector.

The geometries of the CPC are specified by the radius of the entrance aperture, *a*, and exit aperture, *a'* (Figure 1–11).

$$\theta = \sin^{-1}\left(a/a'\right), \tag{1-15}$$

$$L = \left(a + a'\right)\cot\theta \tag{1-16}$$

where θ is the half angle of the geometric field of view, and L is the length of the PCP. The focal length of the paraboloid is computed,

$$f = a'\left(1 + \sin\theta\right) \tag{1-17}$$

in order to calculate the radius, *r*, along the axis of rotation, *z*, to use in MATLAB's *cylinder* function,

$$r = \left(4f\left(z + f\right)\right)^{1/2}. \tag{1-18}$$

Figure 1–12 shows an example parameterized PCP with an entrance aperture of 2mm and an exit aperture of 0.4mm.

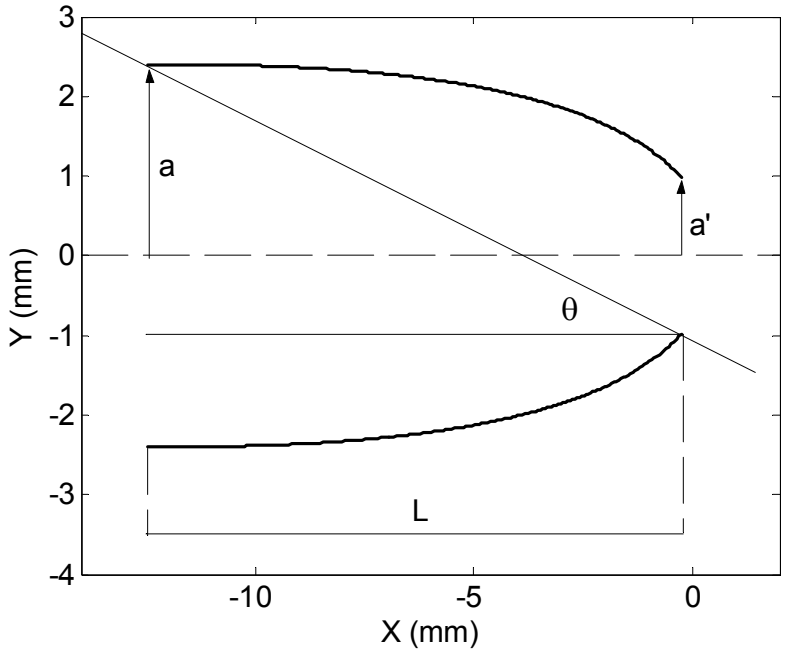

Figure 1–11. Schematic of the CPC denoting *a*, entrance aperture half height; *a'*, exit aperture half height; *L*, length; and, θ, half angle of geometric field of view.

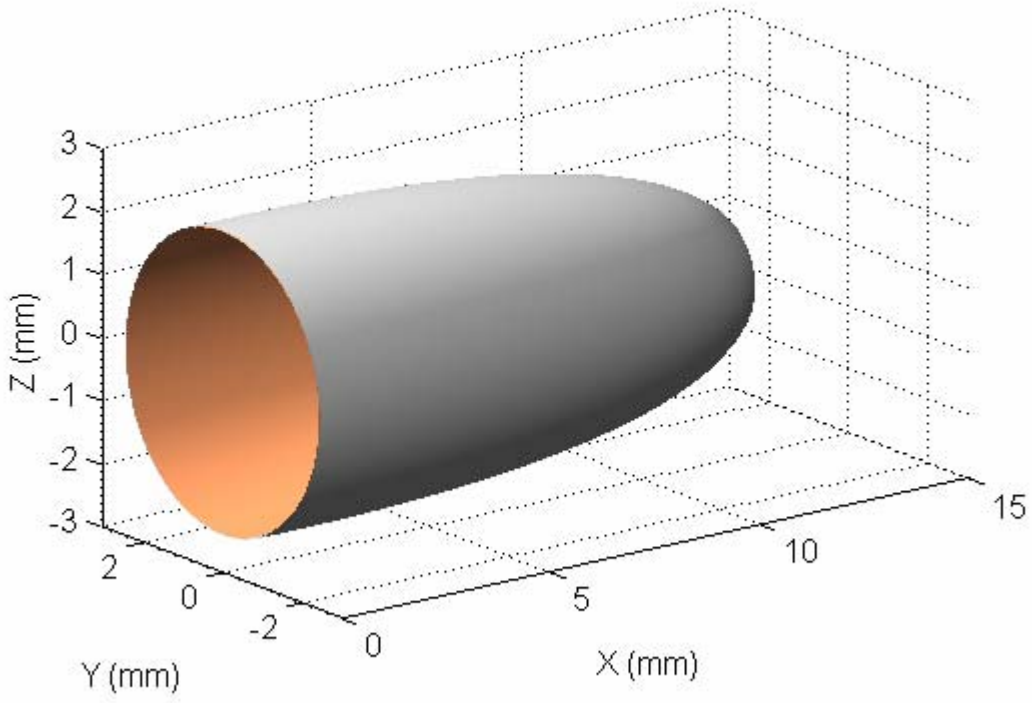

Figure 1–12. Compound parabolic concentrator modeled in MATLAB.

1.7.2. Monte Carlo simulations

The Monte Carlo simulations of rays traveling through the HW setup originate with a ray launched from a random location from within the defined source. The source in the simulation is a 40mm diameter collimated beam originating at 200mm in the y direction and centered on the center of the first OAP mirror. The ray is projected in direction of the first OAP mirror. The nearest vertex on the parameterized surface is determined and the reflected ray is calculated using

$$R = I - 2(N \bullet I)N . \tag{1-19}$$

where N is the vector of the normal at the vertex, I is the vector of the incoming ray and R is the vector of the reflected ray (Figure 1–13).

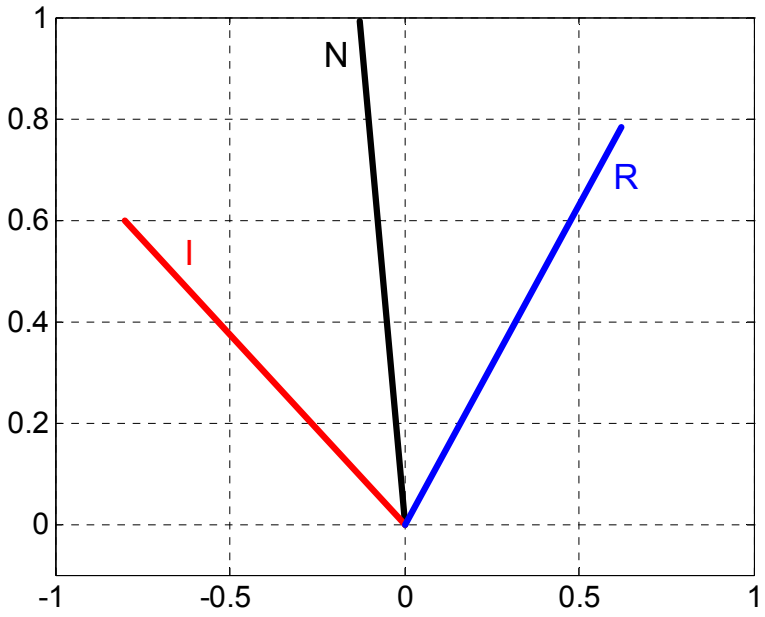

Figure 1–13. Diagram of an incoming ray (I) being reflected (R) through the normal (N) at the point of reflection.

From the first OAP mirror reflection vertex, the position of the ray is calculated for the entrance of the HW. If the yz coordinates are within the opening of the HW, propagation is continued to the first intersection with the interior surface of the HW. If the ray misses the HW opening, the trace is aborted and a new ray is launched from the source. The

intersection point on the interior of the HW is calculated by solving the cylinder equation in the yz plane,

$$\left(y-y_0\right)^2+\left(z-z_0\right)^2-r^2=0\,. \tag{1-20}$$

with the ray equation,

$$P(t)=S+tD\,. \tag{1-21}$$

where *P(t)* is the point in space at time *t* on the ray, *S* is the ray's starting point and *D* is the vector defining its direction. The substitution yields a quadratic equation in *t*. Using the quadratic equation, two solutions for *t* are calculated. The t that progresses the ray in the positive direction is chosen. The reflection is calculated at the closest vertex of the parameterized surface and the process is repeated until the ray reaches the exit of the HW.

In a similar fashion to determining the intersection point on the cylinder, an initial guess of the intersection point on the second OAP mirror is determined by solving the mirror surface quadratic equation for the ray equation in the yz plane. The closest vertex to the ray intersection point is determined using Simplex optimization function included in MATLAB. The cost function to minimize is t value for the intersection of a line and a plane,

$$t=\frac{-\left(S-M\right)\cdot N}{D\cdot N}\,. \tag{1-22}$$

where *S* and *D* are the ray's starting point and directional vector, *M* is the mirror vertex, and *N* is the normal at *M*. The ray is reflected through mirror 2 and project to mirror 3. The projection to OAP mirror 3 is achieved in the same way except in the xz plane. The

final reflection is toward the detector. The ray's point is calculated at x position of the detector and is tested to see if it falls with the yz coordinates of the detector.

In the case of the CPC, intersection is computed by first calculating all the ray's points for the x coordinates of the curve that defines the CPC. The yz coordinates along the ray are used to calculate the distance above the axis of rotation and the point that intersects the CPC curve is determined. The ray's reflection is calculated and the process is repeated until the ray hits the detector at the exit aperture of the CPC or the ray reflects away from the exit aperture.

Simulations are performed in 2-D to determine the affect of the x position of mirror 1 and mirror 2, x and y position of mirror 1, x position and entrance aperture size of the CPC, and CPC entrance and exit aperture size when the CPC is placed 10mm from the exit of the HW. Simulations are performed in 3-D to determine the affect of y and z position of mirror 1, x position and entrance aperture size of the CPC, and CPC entrance and exit aperture size when the CPC is placed 10mm from the exit of the HW. All simulations are performed with the HW accepting all incoming rays and when rays greater than an acceptance angle of 15° are rejected. For each parameter configuration, 1000 rays are launched and the percentage hitting the detector, throughput, and the average distance the good rays travel through the HW are reported.

1.8. References

[1] Baker, R. R., in *Smoke Chemistry, in Tobacco Production, Chemistry and Technology*, Davis, D. L., Nielson, M. T., Eds.; Blackwell Science: London, **1999**; pp 398-439.

[2] Baker, R. R., "Variation of the gas formation regions within a cigarette combustion coal during the smoking cycle." *Beitr. Tabakforsch. Int.*, **1981**, *11*, 1-17.

[3] Baker, R. R., "Variation of sidestream gas formation during the smoking cycle." *Beitr. Tabakforsch. Int.*, **1982**, *11*, 181-93.

[4] Perfetti, T. A., Coleman, III, W. M., Smith, W. S., "Determination of mainstream and sidestream cigarette smoke components for cigarettes of different tobacco types and a set of reference cigarettes." *Beitr. Tabakforsch. Int.*, **1998**, *18*, 95-113.

[5] *Strategy of Experimentation*, Edition 4.2, E.I. du Pont de Nemours and Company, Wilmington, **1988**.

[6] Clement, R. E., Yang, P. W., Koester, C. J., "Environmental Analysis." *Anal. Chem*., **2001**, *73*, 2761-2790.

[7] Koester, C. J., Simonich, S. L., Esser, B. K., "Environmental Analysis." *Anal. Chem*., **2003**, *75*, 2813-2829.

[8] Workman, Jr., J., Creasy, K. E., Doherty, S., Bond, L., Koch, M., Ullman, A., Veltkamp, D. J., "Process analytical chemistry." *Anal. Chem*., **2001**, *73*, 2705-2718.

[9] Chu, P. M., "Advances in optical methods for trace gas analysis." *Anal. Bioanal. Chem.*, **2003**, *376*, 305-307.

[10] DeForest, C. L., Qian, J., Miller, R. E., "Composition Determination of Multicomponent Organic Aerosols by On-Line FT-IR Spectroscopy." *Appl. Spectrosc*., **2002**, *56*, 1429-1435.

[11] Chaffin, Jr., C. T., Marshall, T. L., Chaffin, N. C., "Passive FTIR remote Sensing of smokestack emissions." *Field Anal. Chem. Technol*., **1999**, *3*, 111-115.

[12] Bak, J., Clausen, S., "FTIR emission spectroscopy methods and procedures for real time quantitative gas analysis in industrial environments." *Meas. Sci. Technol.*, **2002**, *13*, 150-156.

[13] Gallagher, N. B., Wise, B. M., Sheen, D. M., "Estimation of trace vapor concentration-pathlengths in plumes for remote sensing applications from hyperspectral images." *Anal. Chim. Acta*, **2003**, *490*, 139-152.

[14] Todd, L. A., Ramanathan, M., Mottus, K., Katz, R., Dodson, A., Mihlan, G., "Measuring chemical emissions using open-path Fourier transform infrared (OP-FTIR) spectroscopy and computer-assisted tomography." *Atmos. Environ.*, **2001**, *35*, 1937-1947.

[15] Dept. of Health and Human Services, "NIOSH pocket guide to chemical hazards." DHHS (NIOSH) Publication No. 97-140, DHHS, Washington, D.C., **2003**.

[16] Occupational Safety & Health Administration, "Table Z-1 limits for air contaminants." http://www.osha.gov/pls/oshaweb/owadisp.show_document?p_table=STANDARDS&p_id=9992, accessed April 27, 2004.

[17] White, J. U., *J. Opt. Soc. Am.*, **1942**, *32*, 285-8.

[18] Marcantili, E. A. J., Schmeltzer, R. A., *Bell Sys. Technol. J.*, **1964**, *43*, 1783.

[19] Saggese, S. J., Harrington, J. A., Sigel, Jr., G. H., Altkorn, R., Haidle, R., "Novel lightpipes for infrared spectroscopy." *Appl. Spectrosc.*, **1992**, *46*, 1194-1197.

[20] Abel, T., Hirsch, J., Harrington; J. A., *Opt. Lett.*, **1994**, *19*, 1034

[21] Saito, M., Nobuyoshi, B., Sawanobori, N., Miyagi, M., "Hollow glass waveguides for mid-infrared light transmission." *Jpn. J. Appl. Phys.*, **1994**, *33*, 164-168.

[22] Matsuura, Y., Abel, T., Harrington, J. A., "Optical properties of small-bore hollow glass waveguides." *Appl. Opt.*, **1995**, *30*, 6842-6847.

[23] Dror, J., Inberg, A., Dahan, R., Elbolm, A., Croitoru, N., "Influence of heating on performances of flexile hollow waveguides for the mid-infrared." *J. Phys. D. Appl. Phys.*, **1996**, *29*, 569-577.

[24] Dahan, R., Inberg, A., Dror, J., Croitoru, N., "Scattering phenomenon investigation of the guiding surface of infrared waveguides for application in medicine." *Proc. SPIE*, **1994**, *2328*, 45-51.

[25] Croitoru, N., Inberg, A., Dahan, R., David, M. B., "Scattering and beam profile measurements of plastic, silica, and metal radiation waveguides." *J. Biomed. Opt.*, **1997**, *2*, 235-242.

[26] Metropolis, N., Ulam, S., "The Monte Carlo method. " *J. Amer. Stat. Assoc.* **1949**, *44*, 335-341.

[27] Winston, R., "Light collection within the framework of geometric optics." *J. Opt. Soc. Amer.*, **1970**, *60*, 245.

[28] Hildebrand, R. H., Winston, R., "Throughput of diffraction-limited field optics system for infrared and millimetric telescopes." *Appl. Opt.*, **1982**, *21*, 1844.

[29] Hildebrand, R. H., Erratum to "Throughput of diffraction-limited field optics system for infrared and millimetric telescopes." *Appl. Opt.*, **1985**, *24*, 616.

[30] Welford, W., Winston, R., *High Collection Nonimaging Optics*, Academic Press, San Diego, **1989**.

CHAPTER 2.

REAL-TIME FT-IR ANALYSIS OF CARBON MONOXIDE AND NITRIC OXIDE IN SIDESTREAM CIGARETTE SMOKE

2.1. Introduction

Various types of infrared spectroscopy have been applied to the analysis of cigarette smoke[1-9]. The majority of these studies focus on the analysis of mainstream cigarette smoke, which comprises smoke emitted from the filter end of the cigarette[1-7,9]. Maddox[3] and Cueto[5,6] demonstrated analysis of gas phase components of mainstream smoke by Fourier transform infrared (FT-IR) spectroscopy. Cole and Martin[8] measured the emission of several gas phase components of sidestream cigarette smoke, i.e. smoke emitted from the lit end of the cigarette, by FT-IR techniques with univariate calibrations. However, their quantitative analysis relied on unique rotation-vibration absorption lines and collecting smoke from four cigarettes simultaneously to compute an average amount emitted per cigarette. Consequently, the FT-IR results revealed significant differences for carbon monoxide (CO) and nitric oxide (NO) in comparison to traditional techniques. Shi, et al.[9] used a quantum cascade laser (QCL) to achieve the desired time resolution for monitoring intra-puff concentration profiles for several gas phase mainstream smoke analytes. The QCL spectrometer was successful in monitoring sidestream smoke, but required independent carbon dioxide measurements obtaining emission ratios and emitted amounts for the gas phase analytes. Clearly, the increased temporal resolution aides in understanding the combustion gas formation during the puff. For sidestream smoke, combustion product emission between products generated from the puff and from

smoldering (inter-puff period) needs to be resolved on the basis of a single cigarette. These results are of increasing importance as new types of cigarette papers are used to meet self-extinguishing standards for "fire safe" cigarettes.

In order to achieve the demanded combination of temporal resolution and accurate yields, FT-IR spectroscopy appears a suitable method collecting data with a fast scanning spectrometer in combination with robust calibrations for quantitative results. Cigarette smoke is known to contain thousands of compounds and the amounts of vapor phase components in the sidestream smoke are emitted at levels two to ten times the mainstream amounts[10,11]. In order to utilize rotation-vibration lines as reported in previous studies[8,9], FT-IR spectra would have to be collected at a spectral resolution of $1cm^{-1}$ or better. Improved temporal resolution of the sidestream concentration profile could only be attained at an equivalent signal-to-noise level by collecting spectra at lower resolution. However, at lower resolution ($>1cm^{-1}$) the rotational bands would no longer be spectroscopically resolved for quantitation.

Multivariate regression techniques, such as partial least squares (PLS), could be used for accurate quantitation in the presence of interfering compounds at unknown levels. Multivariate calibrations are more robust to interferences and in dynamic systems because they use more spectral information for quantitation. FT-IR spectroscopy and PLS have been applied to multi-component analysis of combustion gases[12-15], process streams[16] and environmental pollutants[17-19]. Hart, et al.[17,18] studied the effect of spectral resolution and regression techniques on the performance of open path FT-IR for monitoring environmental pollutants. Qin and Cadet[16] evaluated the combination of low

resolution FT-IR spectra and PLS to characterize effluent from a high temperature process containing organic compounds. Jaakkola, et al.[20] studied the relationship between FT-IR instrument resolution and analytical figures of merit for the analysis of hydrocarbon mixtures. Each of these previous studies achieved successful quantitation at lower spectral resolution. The present study investigates the application of low resolution fast scanning FT-IR spectroscopy for monitoring of carbon monoxide and nitric oxide in sidestream cigarette smoke and compares univariate to multivariate calibration techniques. Optimization of the resolution will be studied as the responsible parameter for reducing the data acquisition time without sacrificing the quantitative analytical performance.

2.2. Experimental Section

All cigarettes used in this study were conditioned and smoked according to ISO guidelines[21,22]. The smoking machine was a five port linear smoking machine with sidestream smoke collection frame (KC Automation, Inc., Richmond, VA) (Figure 2–1). The smoking parameters were a 35mL, 2s puff repeated every 60s. The puff profile was sinusoidal. A 44mm glass fiber filter pad (pore size approx. 1μm) was used to collect the mainstream and sidestream particulate matter.

Five cigarettes were selected for this study. Three were reference cigarettes, Kentucky Reference cigarette 1R4F (KY1R4F), Industry Monitor 16 (IM16), and CORESTA Monitor 3 (CM3). The other two samples were commercial cigarettes, designated A and B, with nominal mainstream total particulate matter (TPM) deliveries of 10 and 2mg/cig., respectively. These five were selected to represent a range of mainstream TPM deliveries

(2 to 20mg), filter ventilation (0 to ~80%), and different blend compositions. Three smoke runs were conducted for each cigarette. Three cigarettes were smoked per run and the FT-IR and analyzer data were collected continuously during the run.

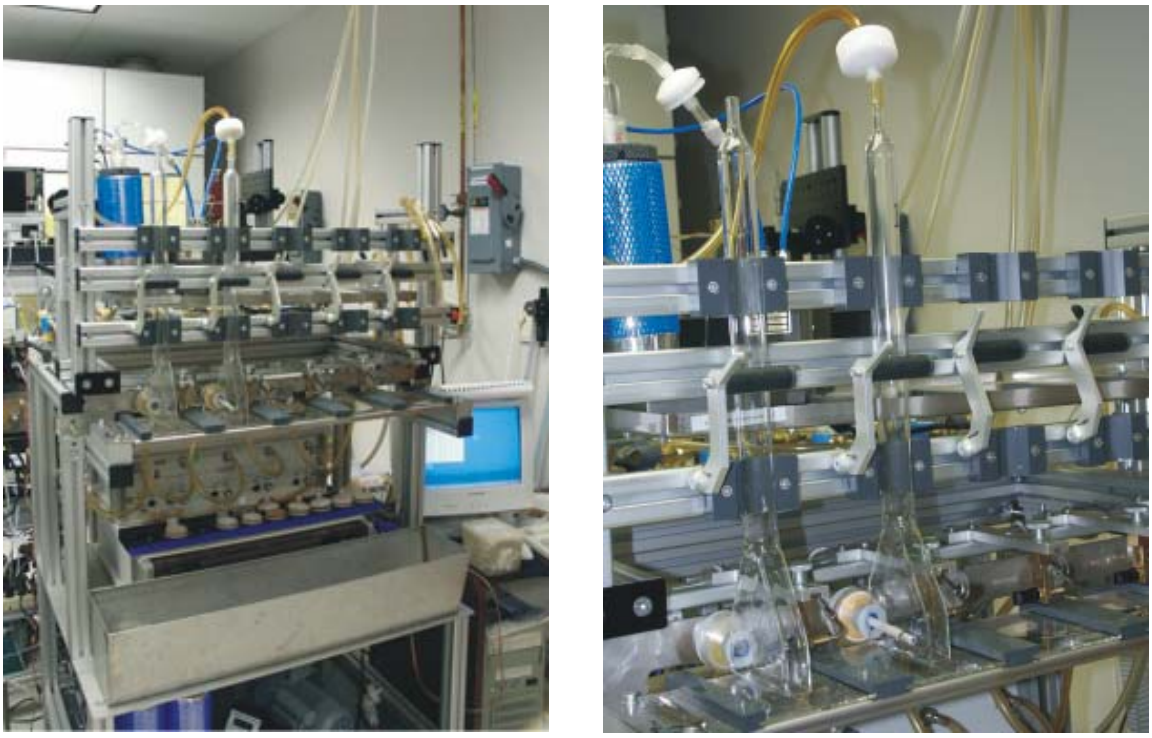

Figure 2–1. Pictures of the five port smoking machine with sidestream collection frame. The complete system is shown on the left. A close-up of the fishtail chimney and a smoldering cigarette is shown on the right.

Figure 2–2 shows a schematic of the sidestream smoke collection apparatus. A fishtail chimney[23] was used to collect the sidestream smoke from the lit end of the cigarette. The flow rate, measured at the top of the chimney was set to 3L/min. The chimney design

permitted cooling of the sidestream smoke and formation of smoke aerosol particles for trapping on the glass fiber filter pad immediately after the chimney. To further reduce the level of volatile and semi-volatile compounds in the sidestream smoke, a cold trap immersed in a dry ice/methanol bath (-78°C) was place immediately after the filter pad. To prevent ice or other solid material from the trap entering the gas cell, another glass fiber filter pad was placed after the trap. After the second filter pad, a metal bellows pump (MB-41, Senior Flexonics, Inc., Sharon, MA) and an in-line flow meter (model 4120A, TSI, Inc., Shoreview, MN) to monitor the sidestream flow rate were inserted.

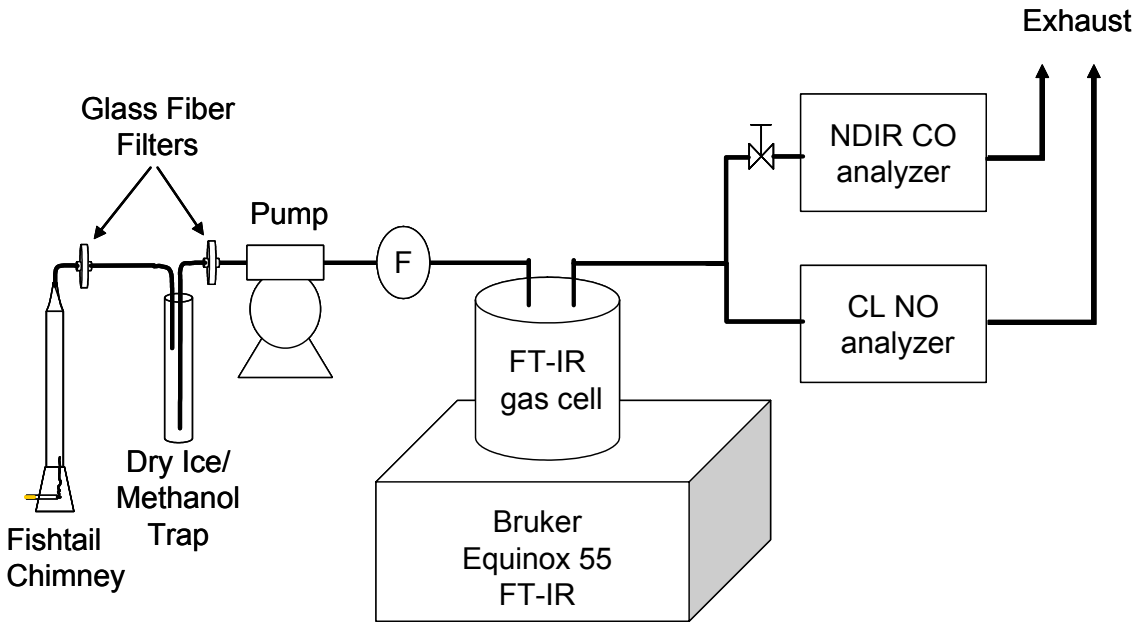

Figure 2–2. Schematic layout of experimental set-up for real time analysis of sidestream cigarette smoke using FT-IR spectroscopy with a multipass gas cell.

Figure 2–3 show the instrumentation used in this study. Data were collected with a Bruker Equinox 55 FT-IR (Bruker Optics, Inc., Bellerica, MA) equipped with a 3m multi-pass gas cell (Gemini Scientific Instruments, Buena Park, CA). The gas cell has an internal volume of 375mL and was heated to 50°C. The FT-IR scan parameters were 64 scans at 100kHz and a spectral resolution of $1cm^{-1}$, yielding a scan time of 21s. Data were collected in the time resolved mode where interferograms are first collected and stored in the memory during the smoke run and processed at the end of the smoke run. The number of scans collected during a smoke run ranged from 80 to 122 depending on the amount of time needed to smoke three cigarettes in succession on the smoke machine. Sample and reference interferograms and absorbance spectra were saved and used to compute sample absorbance spectra at spectral resolutions of 2, 4, and $8cm^{-1}$. The FT parameters for all data were the Norton-Beer apodization (medium), 2x zero filling, and Mertz phase correction.

A non-dispersive infrared (NDIR) CO gas analyzer (ZRH2, California Analytical Instruments, Inc., Orange, CA) and a chemiluminescence (CL) NO gas analyzer (CLD400, California Analytical Instruments, Inc., Orange, CA) were located downstream of the 3m gas cell. The NO analyzer had a pressure regulated flow controller that limited the flow to approximately 2L/min. The analyzers were placed in parallel to each other in the sidestream flow. A needle valve was inserted at the entrance to the CO analyzer to control the total sidestream flow. Analog signals from the two gas analyzers and the flow meter were connected to analog-to-digital inputs of a motor controller board (ESP6000, Newport, Inc., Irvine, CA) and converted to digital values at the rate of one data point per

second for storage and analysis. The data were automatically collected and stored using MATLAB (Mathworks, Inc., Natick, MA). Gas analyzer data were converted from recorded volts to concentration values and finally to the component amount in MATLAB. The amounts per second were summed up over the smoke run and divided by the number of cigarettes smoked to report yields per cigarette.

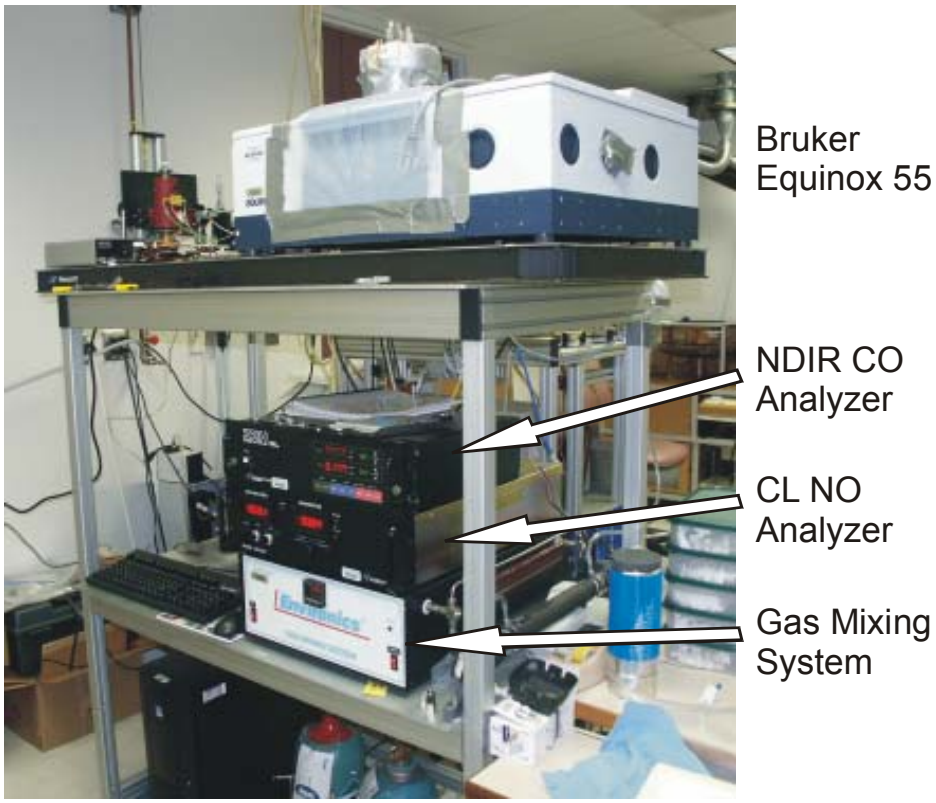

Figure 2–3. Picture of the instrumentation used for the sidestream analysis.

2.2.1. Calibration Methodology

The FT-IR was calibrated using a series of gas standards generated by a gas mixing system (Environics 4000, Environics, Inc., Tolland, CT). The computer controlled gas mixing system used mass flow controllers to meter in calibration gases and dilute them with instrument air. Two calibration gases, 1.01% CO and 600ppm NO with nitrogen balance (Air Products, Inc., Theodore, AL), were applied, which had a certificate of analysis of 2% accuracy. A series of 15 standards was prepared by specifying the concentrations of each analyte and having the software determine the amount of diluent air to add.

In this study, fifteen standards were generated using the "minimum correlation table". Each standard generated from the gas mixing system was collected in a 10L polypropylene gas bag (KC Automation, Inc., Richmond, VA). The gas bag was attached to the first filter pad holder and the calibration gas was introduced to the system in a similar manner as the sidestream smoke. The system was purged for 60s with the gas standard prior to FT-IR data collection. Three FT-IR spectra versus a nitrogen background and the analog output of the analyzers were collected for each standard. The analyzer output was converted to concentration and smoothed using a moving average digital filter with a window size equal to the interferogram scan time. Each calibration gas spectrum was assigned the CO and NO concentration from the smoothed data of the analyzers at the data collection end time of the FT-IR.

Calibrations and subsequent data analysis were performed within three different spectral regions and by both univariate and multivariate calibration approaches. Cole and

Martin[8] performed their analysis of NO and CO using spectral regions containing unique IR rotation-vibration lines and univariate calibrations. The present study evaluated the rotation-vibration lines, and also included the R-branch and the entire absorption band for the CO and NO bands centered at 2144 and 1875cm^{-1}, respectively.

Two different regressions were used in this study. A univariate regression performed a least squares regression of concentration against the integrated absorbance within the evaluated spectral region. The multivariate regression used a PLS approach. As there was no interference between the investigated components, each component was quantitated individually in both cases. The regression programs were part of the OPUS FT-IR software (Bruker Optics, Inc., Bellerica, MA).

2.2.2. Calculation of Yield and Data Analysis

The FT-IR data were analyzed for each smoke run. For a given calibration, the concentration of each analyte was determined for each spectrum in the time resolved experiment. Since the interferograms were co-added and averaged, the resulting concentration represented the average concentration of the gas stream flowing through the gas cell during the collection interval of the interferograms:

$$C_{i,k} = \frac{\int_{t=t_{start}}^{t=t_{end}} c_{i,k}\,\mathrm{d}t}{t_{end} - t_{start}} \qquad (2\text{-}1)$$

where $C_{i,k}$ is the average concentration of component i when interferogram k was collected, $c_{i,k}$ is the concentration of component i in the cell at time dt, and t_{start} and t_{end} are the start and end times for collecting interferogram k.

The concurrent flow rate data were smoothed with a moving average digital filter with a window size equal to the scan time of the interferogram t_{scan}. Therefore, at time t_{end}, flow rate F_k is the average flow rate when interferogram k was collected. The mass of component i for interferogram k, $M_{i,k}$, was calculated by

$$M_{i,k} = \frac{C_{i,k} W_i F_k t_{scan}}{1000 V_m} \tag{2-2}$$

where W_i, and V_m are the molecular weight and the molar volume of component i at room temperature, respectively.

The masses of each component were summed up over the entire smoke run (l interferograms) and divided by the number of cigarettes smoked during the run, n, to report yields per cigarette, Y_i.

$$Y_i = \frac{\sum_{k=1}^{l} M_{i,k}}{n} \tag{2-3}$$

Averages and standard deviations of the yields for each cigarette type from the analyzers and all FT-IR calibrations and resolutions were calculated. Because the analyzer and FT-IR data were collected concurrently, two statistics were used to directly compare FT-IR calibration results to analyzer results. The t-statistic was calculated from the difference of each pair of results,

$$t_i = \left|\bar{d}_i\right| m^{1/2} s_{d_i}^{-1} \tag{2-4}$$

where $\bar{d}_i$ and s_{d_i} were the mean and standard deviation of paired data of component i, and m was the number of smoke runs. The root mean squared error of prediction (RMSEP) was calculated using

$$RMSEP = \left(m^{-1} \sum_{j=1}^{m} \left(Y_{i,j,\text{FT-IR}} - Y_{i,j,\text{analyzer}} \right)^2 \right)^{½} \quad (2\text{-}5)$$

where m was the number of smoke runs and $Y_{i,j,\text{FT-IR}}$ and $Y_{i,j,\text{analyzer}}$ were the yields of component i per cigarette for smoke run j for a FT-IR calibration and the analyzer, respectively.

2.3. Results and Discussion

The use of a cold trap is different from previously reported experimental set-ups for the measurement of gas phase sidestream smoke constituents. Given the reactivity of NO, spectra of a 400ppm NO gas standards were collected with and without the trap in place (Figure 2–4). It is evident that the addition of a cold trap does not change the absorption intensity of the NO standard. However, the quality of the recorded spectra is clearly improved as spectral absorptions of interfering components are efficiently removed.

Therefore, results determined here are comparable to previously published results. The infrared spectra of sidestream cigarette smoke with and without the cold trap are shown in Figure 2–5. The spectrum without the trap is dominated by water vapor and CO_2. With the trap in place the quality of the spectrum is substantially improved. Most of the water and semi-volatile components of sidestream smoke are removed, while the vapor phase components of interest, such as CO and NO, are clearly evident (Figure 2–6). In addition to reducing the complexity of the spectra by removing the semi-volatiles, the optics of the multi-path gas cell are protected and the gas cell can be operated at lower temperatures.

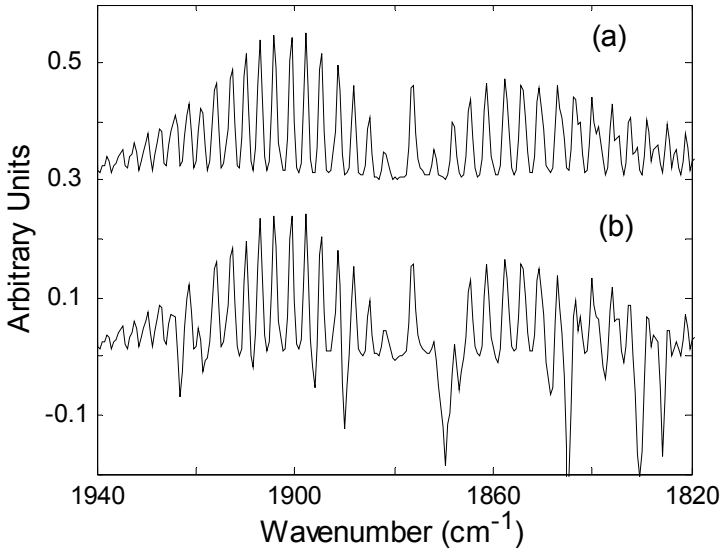

Figure 2–4. Infrared spectra of 400ppm NO gas standard, (a) with and (b) without dry-ice/methanol cold vapor trap.

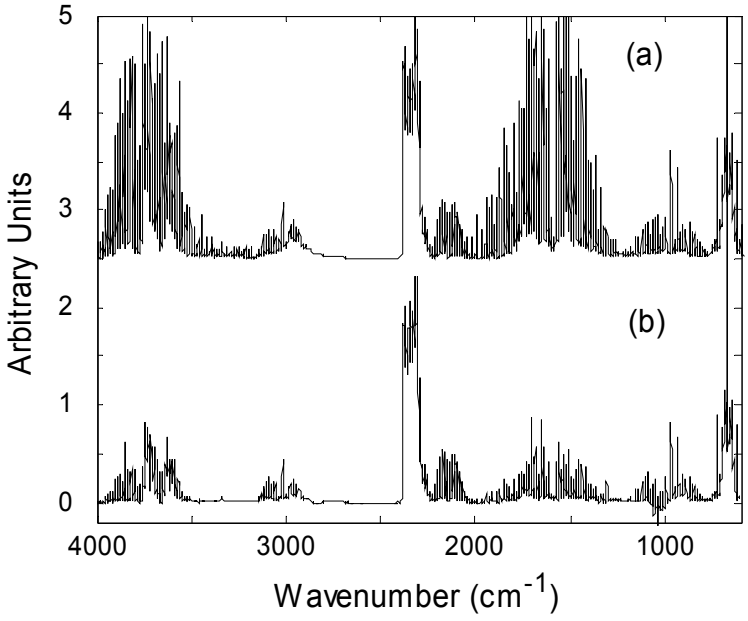

Figure 2–5. Infrared spectra of sidestream cigarette smoke, (a) without and (b) with dry-ice/methanol cold vapor trap.

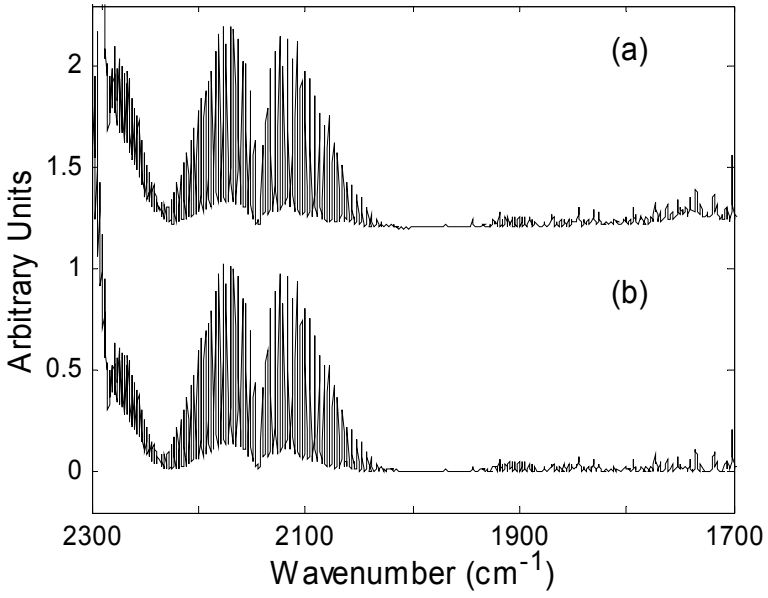

Figure 2–6. Comparison of infrared spectra of (a) sidestream cigarette smoke and (b) a calibration standard containing 2183ppm CO and 65ppm NO.

Loss of rotation-vibration line resolution is clearly evident in the spectra with a resolution of $4cm^{-1}$. Consummate with reduction in rotation-vibration line resolution is reduction in absorption intensity. The spectrum with a resolution of $8cm^{-1}$ has a maximum absorption that is 33% of the maximum absorption in spectrum recorded with a resolution of $1cm^{-1}$. Jaakkola, et al.[20] reported the lower intensity at lower resolution as an advantage, even though it is a source of non-linearity for Beer's law[24]. By using multivariate regression techniques like PLS, the effect of the non-linearity can be minimized and full advantage taken of the increased dynamic range at lower resolution.

Figure 2–7 shows the real time trace of the NO concentration for a smoke run of the KY1R4F. Evident are breaks between the three cigarettes smoked during the run and the

slight but noticeable increases in concentration due to puffing. The temporal resolution achieved in these measurements is a substantial improvement from previous works based on FT-IR[8], better resembling the results attained with the analyzer. The FT-IR trace provides less temporal resolution than the analyzer trace due to a 21s sampling interval versus 1s provided by the analyzer. The scan time of the FT-IR at 21s is a limitation due to the volume and flow response time of the gas cell. The variability during the smoke run demonstrates the need to perform real time analyses for gas phase smoke constituents with sufficient temporal resolution in order to measure inter-puff concentrations.

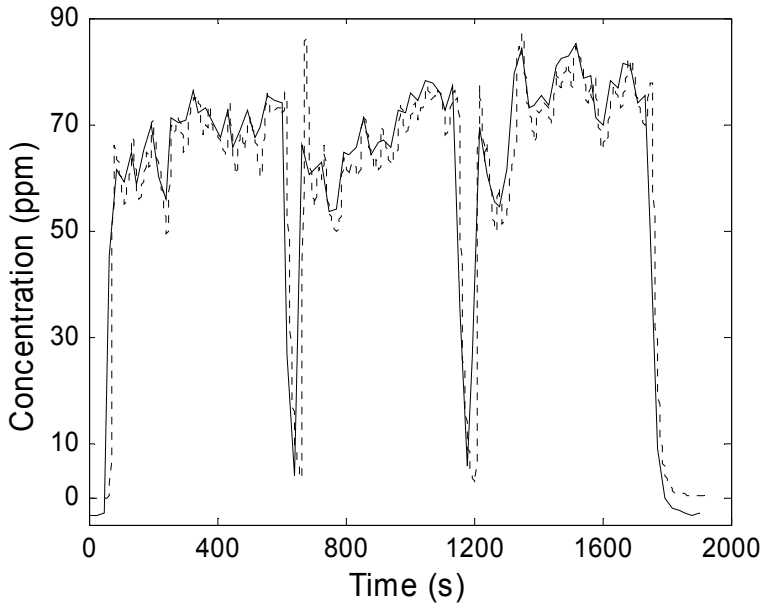

Figure 2–7. Nitric oxide concentration trace for a Kentucky 1R4F reference cigarette smoke run (··· NO analyzer, — FT-IR using PLS calibration).

FT-IR performance will continue to approach the required temporal resolution with the advent of faster scanning instruments and gas cells with faster flow responses, such as hollow waveguides as shown in Chapters 3 and 4.

In sidestream cigarette studies, the total yield for the smoke run is the integrated area of the time versus concentration trace converted into constituent mass. The sample results are reported in mass per cigarette units. However, this complicates the optimization and validation of PLS regression models. Selecting the rank in the PLS calibrations based on yields and not concentration results for individual spectra produces different results. Figure 2–8 through Figure 2–11 show the effects of PLS latent variable selection on RMSEP and root mean square error of cross validation (RMSECV) in the application of measuring CO and NO in sidestream cigarette smoke. The RMSEP is calculated from the difference in yields between the FT-IR and the analyzer results. The RMSECV values are calculated during the PLS calibration process and have the same units as the standards.

The FT-IR PLS calibration usually recommends using the number of latent variables with the smallest predicted residual sum of squares (PRESS). However, the recommended number of latent variables based on the lowest RMSEP was different than the number recommended from the PRESS for 75 and 83% of the CO and NO calibrations, respectively (Table 2–1). In the case of CO calibrations using either the R-branch or the entire band, the optimum number of latent variables was less than or equal to the recommended number. For example, using spectra with a resolution of $4cm^{-1}$, the optimum number of latent variables was one. This seems counter-intuitive to the notion

that more latent variables are needed to model noise and interferences for PLS regressions of a single component[25] and with the array of components present in sidestream cigarette smoke, this would appear desirable. Figure 2–12 shows the spectral loadings for the first five latent variables from the calibration using the entire band at a spectral resolution of $4cm^{-1}$.

Table 2–1. Selected PLS latent variables.

Resolution	Spectral Region	CO		NO	
		RMSECV	RMSEP	RMSECV	RMSEP
1	Rotational	9	2	2	3
	R-branch	3	3	5	5
	Entire band	3	3	5	8
2	Rotational	10	10	5	4
	R-branch	3	1	5	6
	Entire band	5	1	6	7
4	Rotational	5	9	5	5
	R-branch	3	1	5	4
	Entire band	4	1	4	9
8	Rotational	5	6	3	4
	R-branch	3	1	4	7
	Entire band	4	1	4	7

Spectral information is evident in the first two spectral loadings for CO. The next three show the appearance of noise. Their contribution results in higher RMSEP of yields. The PLS calibrations for NO require more latent variables due to interferences from rotational bands of water and reduced signal-to-noise levels at lower absorption. The NO spectral loadings (Figure 2–12) show the influence of NO and water in the first and

second latent variables, but do not have the appearance of noise until the fifth latent variable. The remaining loadings have the appearance of noise. Their contribution results in higher RMSEP of yields.

The use of only one latent variable in lower resolution CO calibrations with larger spectral regions demonstrates the lack of noise or interferences present in that spectral region. Table 2–2 lists the results for CO calibrations. At lower resolution, the PLS calibrations using either the R-branch or the entire band provide nearly identical results and are close to the univariate results for the same spectral range. Using a resolution of $8cm^{-1}$ would reduce the scan time by a factor of 8. Therefore, a combination of low resolution and 16 scans would produce one spectrum per second. This would provide sufficient time resolution to evaluate the sidestream smoke generated during the inter-puff smoldering period without the complexity or cost of the QCL spectrometer system reported by Shi, et al.[9].

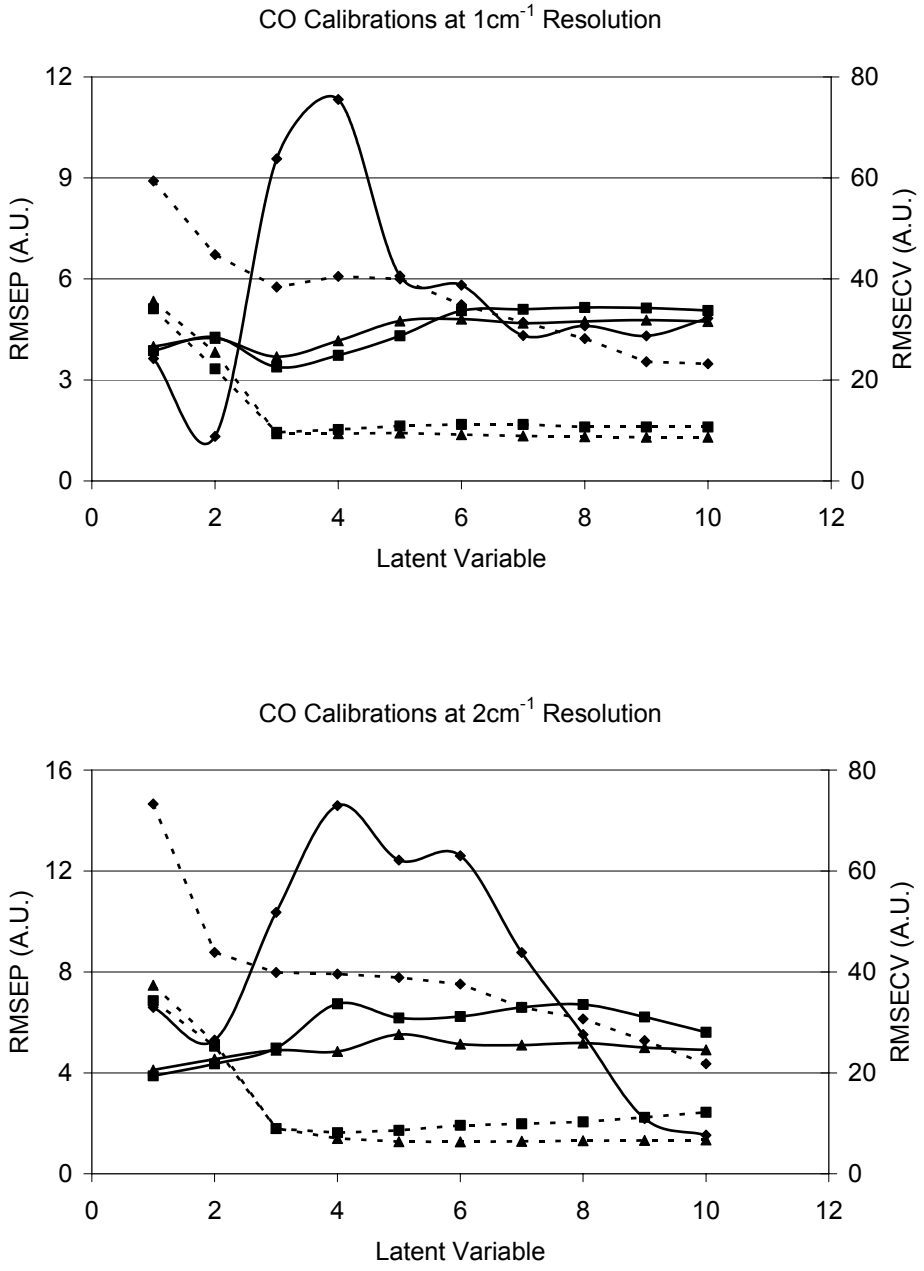

Figure 2–8. Plots of the effect of PLS latent variables on RMSEP and RMSECV for CO cigarette yields at 1 and 2cm^{-1} spectral resolution.
(Spectral region: ◆ Rotational band, ■ R branch, ▲ Entire band,
Statistic: — RMSEP, – – RMSECV)

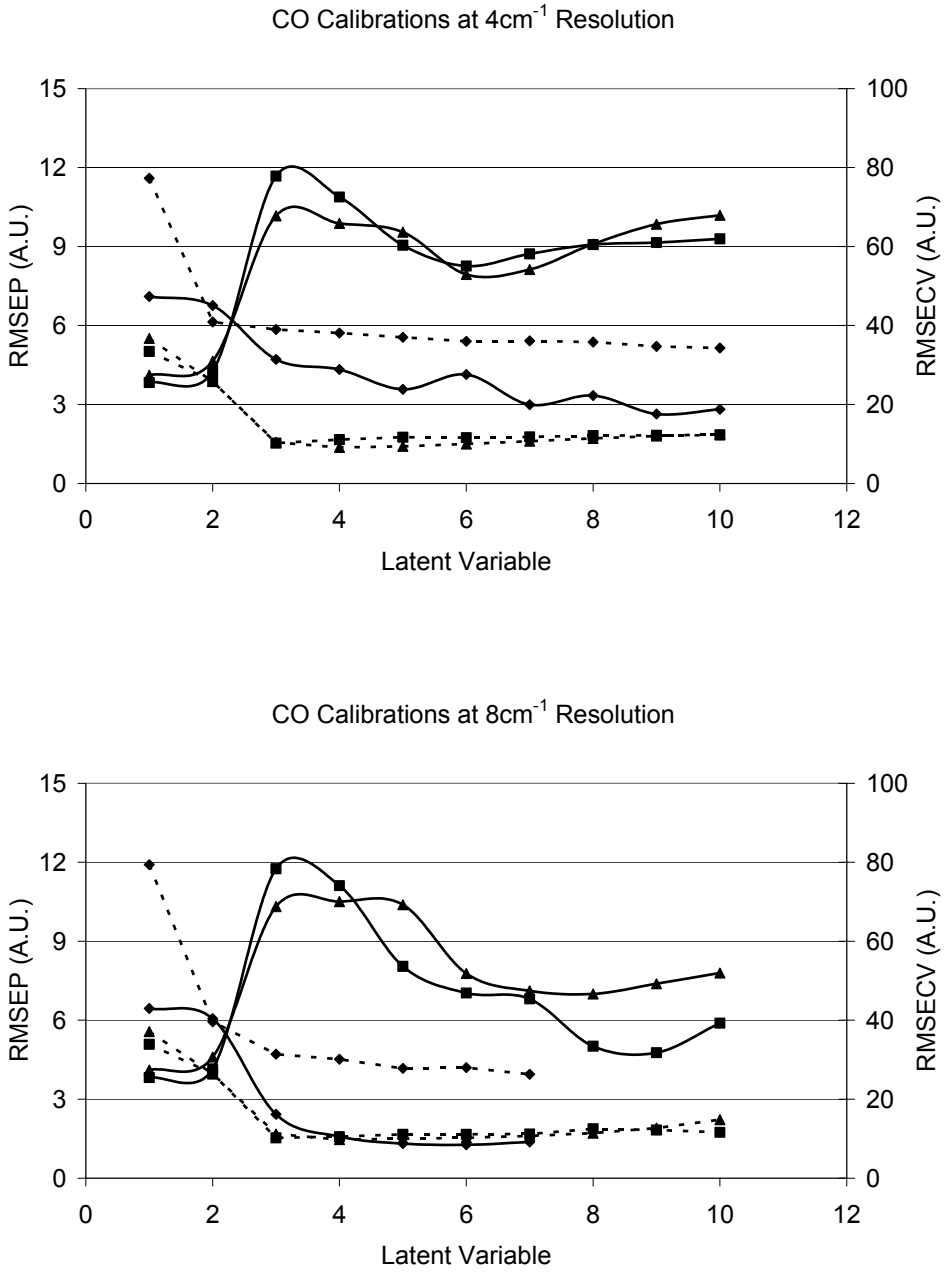

Figure 2–9. Plots of the effect of PLS latent variables on RMSEP and RMSECV for CO cigarette yields at 4 and 8cm^{-1} spectral resolution.
(Spectral region: ◆ Rotational band, ■ R branch, ▲ Entire band,
Statistic: — RMSEP, – – RMSECV)

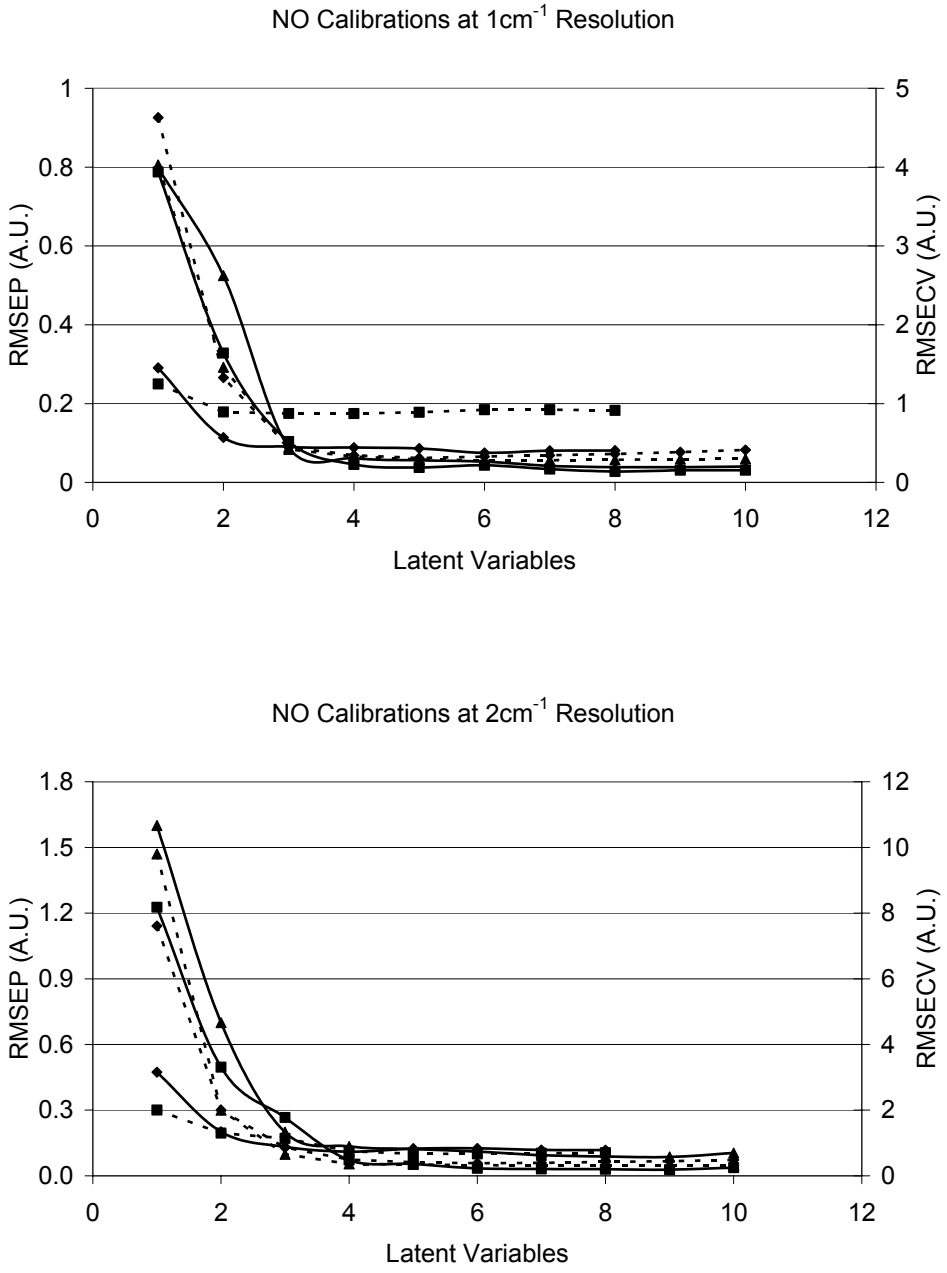

Figure 2–10. Plots of the effect of PLS latent variables on RMSEP and RMSECV for NO cigarette yields at 4 and 8cm^{-1} spectral resolution.
(Spectral region: ◆ Rotational band, ■ R branch, ▲ Entire band,
Statistic: — RMSEP, – – RMSECV)

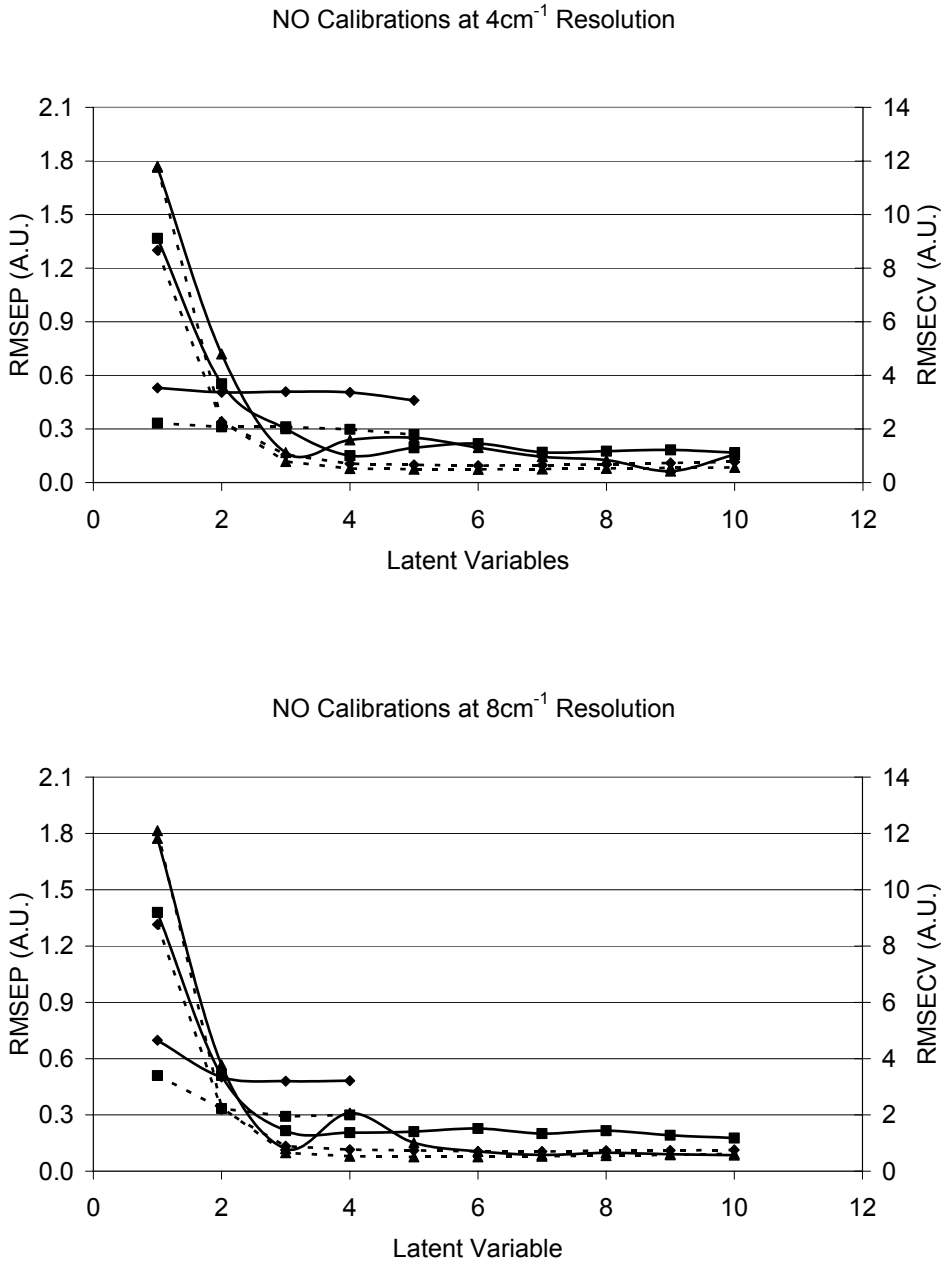

Figure 2–11. Plots of the effect of PLS latent variables on RMSEP and RMSECV for NO cigarette yields at 4 and 8cm^{-1} spectral resolution.
(Spectral region: ◆ Rotational band, ■ R branch, ▲ Entire band,
Statistic: — RMSEP, – – RMSECV)

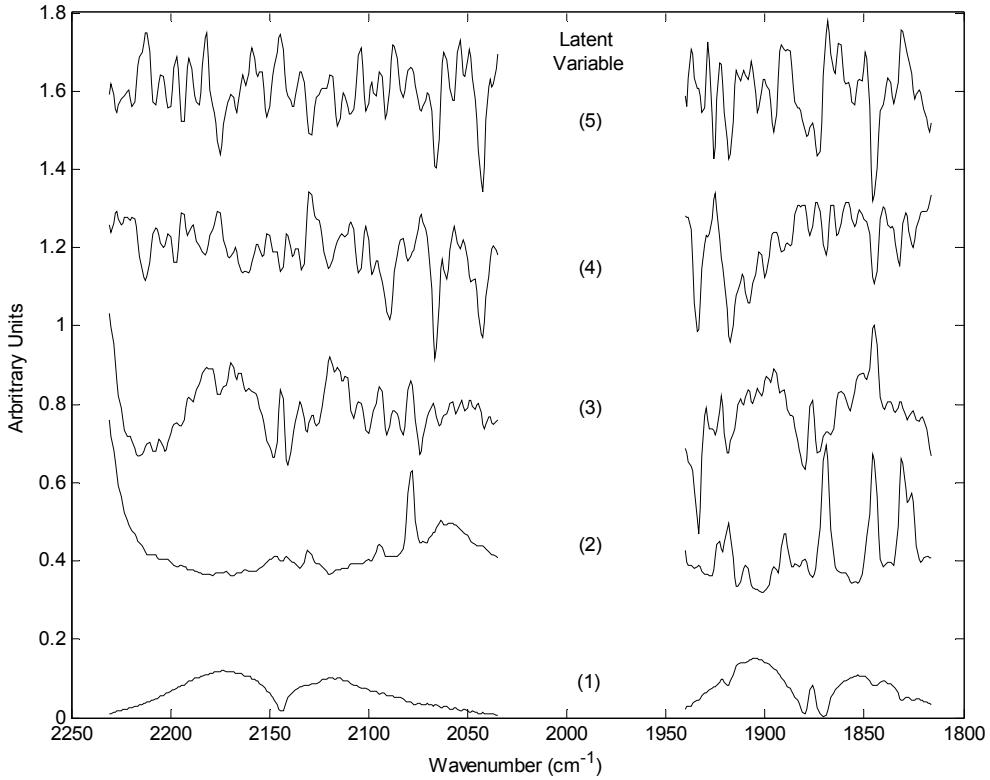

Figure 2–12. Spectral loadings for the first five PLS latent variables of the CO and NO regressions using the entire band at 4cm^{-1} spectral resolution.

The obtained results show that almost all FT-IR calibrations underestimate the CO yield in comparison to the yields obtained by NDIR measurements. For most of the calibrations the FT-IR underestimates the CO analyzer by less than 10%. The instances when FT-IR measurements overestimate the CO yield, the calibrations exclusively used the rotation-vibration line spectral range. The relative uncertainties in the CO results are consistently approx. 7% for all calibrations except for the measurements at a resolution of $4cm^{-1}$ and univariate regression using the rotation-vibration line spectral region. The calibration combination of PLS regression using the rotation-vibration line spectral region has the lowest average RMSEP and has relative uncertainties ranging from 2.1 to 12.7% for the FT-IR versus 4.8 to 9.9% for the CO analyzer.

The PLS calibrations for NO require more latent variables due to interferences from rotation-vibration lines of water and reduced signal-to-noise levels at lower absorbance. The NO spectral loadings show the influence of NO and water in the first and second latent variables, but do not have the appearance of noise until the fifth latent variable. Table 2–3 lists the results for the NO calibrations. The univariate FT-IR calibrations tend to overestimate the NO yield measured by the CL analyzer, whereas the PLS calibrations show good agreement. The relative uncertainty is not consistent for the NO calibrations. On average, relative uncertainties in the PLS results are lower by approx. 19%, compared to approx. 30% for univariate results. This is in contrast to 8.7% for the NO analyzer. The PLS calibration using the entire band has an average relative uncertainty of 9.6%, demonstrating comparable performance with the NO analyzer.

Table 2–2. Results for CO (mg/cig.) for five different cigarettes tested. For PLS calibrations, the number of latent variables used was based on lowest RMSEP.

Resolution	Regression	Spectral Range	Cigarette				
			CM3	KY1R4F	IM16	A	B
CO Analyzer			62.4 ± 4.2	66.6 ± 3.2	57.6 ± 4.5	49.5 ± 4.0	57.9 ± 5.7
1	Univariate	Rotational	61.7 ± 4.0	65.4 ± 2.2	57.6 ± 2.5	49.0 ± 3.5	57.1 ± 5.0
		R-branch	58.5 ± 3.2	61.4 ± 3.2	53.8 ± 3.5	45.8 ± 3.7	54.1 ± 4.7
		Entire	58.0 ± 3.0	60.4 ± 3.5	53.0 ± 3.5	45.3 ± 3.7	53.5 ± 5.2
	PLS[a]	Rotational	61.4 ± 2.7	67.6 ± 2.7	58.7 ± 2.2	49.4 ± 2.2	56.1 ± 6.2
		R-branch	58.7 ± 5.0	63.5 ± 3.0	55.0 ± 0.5	46.4 ± 4.2	55.9 ± 5.7
		Entire	58.3 ± 5.5	63.3 ± 3.0	54.7 ± 0.5	46.0 ± 4.7	55.8 ± 5.7
2	Univariate	Rotational	66.6 ± 5.0	70.3 ± 0.7	62.2 ± 0.7	54.4 ± 2.7	60.7 ± 3.7
		R-branch	58.9 ± 3.5	61.5 ± 3.2	53.9 ± 3.5	46.0 ± 3.5	54.3 ± 5.0
		Entire	58.4 ± 3.2	60.2 ± 3.5	52.8 ± 3.5	45.3 ± 3.2	53.7 ± 5.2
	PLS[a]	Rotational	60.7 ± 3.7	65.9 ± 2.5	57.9 ± 4.7	48.9 ± 5.0	55.2 ± 6.0
		R-branch	58.0 ± 3.7	63.3 ± 2.7	55.1 ± 1.2	45.9 ± 2.5	53.6 ± 5.7
		Entire	57.8 ± 4.2	63.1 ± 2.7	54.9 ± 1.5	45.7 ± 2.7	53.4 ± 5.7
4	Univariate	Rotational	20.8 ± 15.7	6.9 ± 7.9	5.0 ± 4.2	9.7 ± 12.4	15.2 ± 7.7
		R-branch	59.1 ± 3.7	61.7 ± 3.2	54.0 ± 3.5	46.1 ± 3.5	54.4 ± 5.2
		Entire	58.6 ± 3.5	60.4 ± 3.2	52.9 ± 3.7	45.5 ± 3.2	53.8 ± 5.5
	PLS[a]	Rotational	61.2 ± 6.5	69.1 ± 2.7	60.0 ± 1.2	49.6 ± 4.5	56.6 ± 7.2
		R-branch	58.0 ± 4.2	63.5 ± 2.7	55.2 ± 1.2	45.9 ± 2.7	53.8 ± 5.7
		Entire	57.7 ± 5.0	63.3 ± 2.7	55.0 ± 1.2	45.6 ± 3.2	53.6 ± 5.7
8	Univariate	Rotational	111 ± 7	126 ± 10	109 ± 5	99.3 ± 8.2	116 ± 13
		R-branch	59.5 ± 3.7	61.9 ± 3.2	54.1 ± 3.5	46.3 ± 3.5	54.7 ± 5.5
		Entire	58.8 ± 3.5	60.5 ± 3.2	53.0 ± 3.7	45.5 ± 3.2	53.9 ± 5.2
	PLS[a]	Rotational	61.5 ± 3.0	66.7 ± 2.5	58.8 ± 4.0	49.7 ± 2.5	55.9 ± 7.0
		R-branch	58.0 ± 4.2	63.6 ± 2.7	55.2 ± 1.2	45.9 ± 2.7	53.8 ± 5.7
		Entire	57.7 ± 5.0	63.3 ± 2.7	55.0 ± 1.2	45.7 ± 3.5	53.6 ± 5.7

Table 2–3. Results for NO (mg/cig.) for five different cigarettes tested. For PLS calibrations, the number of latent variables used was based on lowest RMSEP.

Resolution	Regression	Spectral Range	Cigarette				
			CM3	KY1R4F	IM16	A	B
NO Analyzer			1.66 ± 0.15	2.31 ± 0.12	1.99 ± 0.20	1.76 ± 0.15	1.91 ± 0.20
1	Univariate	Rotational	1.50 ± 0.25	2.24 ± 0.32	1.91 ± 0.17	1.66 ± 0.25	1.77 ± 0.15
		R-branch	2.96 ± 0.65	3.23 ± 1.49	2.75 ± 0.27	3.46 ± 0.72	3.76 ± 1.44
		Entire	2.99 ± 0.65	2.86 ± 1.32	2.45 ± 0.20	3.25 ± 0.92	3.84 ± 1.61
	PLS[a]	Rotational	1.64 ± 0.97	2.58 ± 0.20	2.11 ± 0.20	1.82 ± 0.82	2.19 ± 0.32
		R-branch	1.65 ± 0.25	2.42 ± 0.15	2.04 ± 0.07	1.78 ± 0.17	2.00 ± 0.22
		Entire	1.61 ± 0.10	2.35 ± 0.15	2.00 ± 0.12	1.74 ± 0.05	1.92 ± 0.20
2	Univariate	Rotational	1.53 ± 0.25	2.26 ± 0.32	1.94 ± 0.15	1.70 ± 0.22	1.81 ± 0.20
		R-branch	2.83 ± 0.60	3.18 ± 1.34	2.74 ± 0.32	3.31 ± 0.62	3.60 ± 1.39
		Entire	2.84 ± 0.62	2.85 ± 1.47	2.43 ± 0.27	3.29 ± 0.92	3.79 ± 1.79
	PLS[a]	Rotational	1.63 ± 0.27	2.41 ± 0.20	2.04 ± 0.12	1.79 ± 0.12	2.03 ± 0.15
		R-branch	1.64 ± 0.07	2.35 ± 0.12	2.02 ± 0.12	1.77 ± 0.10	1.92 ± 0.20
		Entire	1.52 ± 0.05	2.28 ± 0.17	1.94 ± 0.12	1.66 ± 0.02	1.83 ± 0.17
4	Univariate	Rotational	2.23 ± 0.52	2.15 ± 0.94	2.00 ± 0.12	1.56 ± 0.45	2.10 ± 1.04
		R-branch	2.52 ± 0.40	3.03 ± 0.92	2.62 ± 0.30	2.95 ± 0.40	3.18 ± 1.09
		Entire	2.10 ± 0.27	2.41 ± 1.07	2.05 ± 0.35	2.56 ± 0.50	2.96 ± 1.39
	PLS[a]	Rotational	1.53 ± 1.79	2.67 ± 0.22	2.12 ± 0.45	1.67 ± 1.52	2.27 ± 0.25
		R-branch	1.78 ± 0.10	2.46 ± 0.10	2.14 ± 0.10	1.89 ± 0.17	2.08 ± 0.25
		Entire	1.57 ± 0.10	2.31 ± 0.20	1.98 ± 0.15	1.69 ± 0.10	1.83 ± 0.15
8	Univariate	Rotational	1.84 ± 1.42	2.04 ± 1.89	1.95 ± 0.17	0.65 ± 1.52	1.55 ± 1.64
		R-branch	2.31 ± 0.20	2.91 ± 0.57	2.53 ± 0.22	2.66 ± 0.17	2.87 ± 0.79
		Entire	1.04 ± 0.32	1.76 ± 0.42	1.47 ± 0.40	1.46 ± 0.15	1.72 ± 0.75
	PLS[a]	Rotational	1.38 ± 2.06	2.55 ± 0.15	1.99 ± 0.52	1.62 ± 1.74	2.12 ± 0.32
		R-branch	1.85 ± 0.62	2.45 ± 0.12	2.17 ± 0.25	1.96 ± 0.65	2.04 ± 0.22
		Entire	1.65 ± 0.55	2.31 ± 0.22	2.02 ± 0.30	1.78 ± 0.52	1.87 ± 0.17

Because of differences in the cigarettes and sample collection, few direct comparisons can be made between these studies and the work of Cole and Martin[8]. The t-statistic for a spectral resolution of 1cm^{-1} (Table 2–4) shows improvements in agreement between the FT-IR and the analyzer results when using PLS. Using the rotation-vibration line spectral region for CO calibrations, there is no significant difference from the NDIR results for measurements at a spectral resolution of 1, 4, or 8cm^{-1}. For NO, the PLS calibrations perform better with no significant difference from the chemiluminescence method when a larger spectral region is used. Surprisingly, the NO calibration with the lowest t-statistic is obtained at a resolution of 8cm^{-1} with PLS calibration using the entire band. Summarizing, the improved performance compared to previous FT-IR results[8] is attributed to the developed experimental set-up (i.e. collecting smoke from one cigarette, a smaller volume gas cell, and concurrent data collection), greater temporal resolution of the smoke run and the application of PLS regression algorithms.

The RMSEPs for the CO and NO calibrations are listed in Table 2–5. The best performing CO calibration is the univariate regression using the rotation-vibration line spectral region of spectra recorded with a resolution of 1cm^{-1}. As expected, due to non-linearities of the Beer-Lambert law at lower spectral resolution[24], the univariate calibration yields larger errors when used at lower resolutions. PLS regression with the same spectral region does not noticeably degrade in performance at lower resolutions, which suggests that lower spectral resolution can be used to increase the temporal resolution of the smoke run traces without sacrificing performance.

Table 2–4. Results of paired t-test for CO and NO in sidestream smoke. With α=0.025 and 14 degrees of freedom the critical t-value is 2.145. For PLS calibrations, the number of latent variables used was based on lowest RMSEP.

Analyte	Calibration Type	Spectral Range	Resolution (cm^{-1})			
			1	2	4	8
CO	Univariate	Rotational	3.58	12.30	17.68	37.28
		R-Branch	23.84	21.34	21.03	19.30
		Entire band	21.90	18.84	18.66	18.28
	PLS[a]	Rotational	0.45	3.74	0.76	0.78
		R-Branch	5.90	9.22	8.29	8.16
		Entire band	6.08	9.25	8.16	8.05
NO	Univariate	Rotational	6.60	5.41	0.87	1.85
		R-Branch	8.97	9.35	10.24	12.23
		Entire band	6.46	6.19	3.87	7.99
	PLS[a]	Rotational	2.87	2.19	1.08	0.08
		R-Branch	0.21	2.10	20.61	6.21
		Entire band	1.36	5.24	4.48	0.05

Table 2–5. RMSEP results. For PLS calibrations, the number of latent variables used was based on lowest RMSEP.

Analyte	Calibration Type	Spectral Range	Resolution (cm^{-1})			
			1	2	4	8
CO	Univariate	Rotational	0.88	4.24	48.32	53.62
		R-Branch	4.13	3.92	3.77	3.56
		Entire band	4.81	4.78	4.63	4.55
	PLS[a]	Rotational	1.32	1.53	2.64	1.26
		R-Branch	3.39	3.88	3.84	3.82
		Entire band	3.69	4.12	4.11	4.11
NO	Univariate	Rotational	0.13	0.09	0.37	0.72
		R-Branch	1.42	1.30	1.00	0.76
		Entire band	1.34	1.31	0.69	0.48
	PLS[a]	Rotational	0.09	0.11	0.46	0.48
		R-Branch	0.04	0.03	0.15	0.20
		Entire band	0.04	0.09	0.06	0.09

This circumstance is surprising for the rotation-vibration line spectral region because of the limited number of data points available for the PLS regression and quantitation at the shoulder of the absorption band. At a resolution of $8cm^{-1}$, the spectral region of the rotation-vibration line has only seven data points. Yet, the PLS regression performs better at a resolution of $8cm^{-1}$ than it does at $1cm^{-1}$. The fact that the PLS regression used 6 latent variables on 7 data points indicates that the PLS regression almost reduces to a multiple linear regression approach. For the larger spectral regions, there is significant bias consistently observed across the regression type and spectral resolution. If the bias could be minimized, then comparable performance would be attained with lower resolution spectra. The univariate NO calibrations using larger spectral regions show large errors due to interferences. Using the unique rotation-vibration line of NO the univariate calibration performs well, but degrades when the rotation-vibration line structure is lost at lower spectral resolutions. A similar trend in RMSEP is seen in the PLS calibration using the rotation-vibration line spectral region. Improved performance is evident using PLS regression with larger spectral regions. These results demonstrate the utility of PLS to compensate for unmodeled interferences in complex spectra of cigarette smoke.

2.4. Conclusions

Improvements to the application of FT-IR spectroscopy for the measurement of CO and NO in sidestream cigarette smoke have been demonstrated. Modifications in the experimental set-up yielded enhanced temporal resolution of the smoke run. The obtained data were able to discriminate smoke generated at the puff from between puffs.

The lower volume gas cell reduces the flow response time of the cell, which complements the faster scanning FT-IR system. The additional modification of inserting a cold trap prior to the FT-IR reduced the level of interfering compounds for the analysis of NO. The in-line configuration of CO and NO analyzers with the FT-IR allowed direct comparison of the performance on the same physical samples.

The application of PLS did improve the prediction capability, however, models should be developed with caution for sidestream smoke analysis. PLS calibrations were developed which had results that were not significantly different from analyzer results with the performance depended on appropriate selection of the evaluated spectral region and the number of latent variables. The key metric for sidestream smoke analysis is the constituent yield. Hence, PLS attribute selection based on concentration values can compromise the performance. In general, PLS calibrations were developed providing better performance than univariate calibrations and could also be developed for lower resolution spectra. The advent of collecting data at lower spectral resolutions can significantly reduce the scan time and improve the temporal resolution of FT-IR for routine sidestream smoke analysis.

2.5. References

[1] Williams, T.B., Belk, C.W., "An infrared method for the determination of carbon monoxide and carbon dioxide levels in cigarette smoke." *Beitr. zur Tabakforsch.*, **1972**, *6*, 210-215.

[2] Vilcins, G., Lephardt, J.O., "Fourier transform infrared spectroscopy applied to cigarette smoke analysis." *Recent Adv. Tob. Sci.*, **1976**, *1*, 123-146.

[3] Maddox, W.L., Mamantov, G., "Analysis of cigarette smoke by Fourier transform infrared spectrometry." *Anal. Chem.*, **1977**, *49*, 331-336.

[4] Williams, T.B., "The determination of nitric oxide in gas phase cigarette smoke by non-dispersive infrared analysis. " *Beitr. zur Tabakforsch. Intl.*, **1980**, *10*, 91-99.

[5] Cueto, R., Church, D.F., Pryor, W.A., "Quantitative Fourier transform infrared analysis of gas phase cigarette smoke and other gas mixtures." *Anal. Lett.*, **1989**, *22*, 751-763.

[6] Cueto, R., "Quantitative FTIR analysis of gas phase cigarette smoke." PhD Dissertation, Louisiana State University and Agricultural and Mechanical College, **1990**.

[7] Parrish, M.E., Lyons-Hart, J.L., Shafer, K.H., "Puff-by-puff and intrapuff analysis of cigarette smoke using infrared spectroscopy." *Vibrational. Spec.*, **2001**, *27*, 29-42.

[8] Cole, S.K., Martin, P., "Determination of gas-phase sidestream cigarette smoke components using Fourier transform infrared spectrometry." *Analyst*, **1996**, *121*, 495-500.

[9] Shi, Q., Nelson, D.D., McManus, J.B., Zahniser, M.S., Parrish, M.E., Baren, R.E.; Shafer, K.H., Harward, C.N., "Quantum cascade infrared laser spectroscopy for real-time cigarette smoke analysis." *Anal. Chem.*, **2003**, *75*, 5180-5190.

[10] Dube, M. F., Green, C.R. *Recent Advances in Tobacco Science*, **1982**, *8*, 42-102.

[11] Baker, R.R. in *Tobacco Production, Chemistry and Technology*, Davis, D.L., Nielson, M.T., Eds.; Blackwell Science, London, **1999**.

[12] Martin, D.M., Medvecz, P.J., Nichols, K.M., "Application of FT-IR absorption spectroscopy for *in situ* gas concentration and temperature measurements in

laboratory- and pilot-scale combustion environments." *Appl. Spectroc.* **1993**, *47*, 1898-1906.

[13] Speitel, L.C. J., "Fourier transform infrared analysis of combustion gases." *Fire Sci.* **2002**, *20*, 349-371.

[14] Blackwood, T.R., "An evaluation of flare combustion efficiency using open path Fourier transform infrared technology." *J. Air Waste Manage. Assoc.* **2000**, *50*, 1714-1722.

[15] Haus, R., Schafer, K., Bautzer, W.; Heland, J., Mosebach, H., Bittner, H., Eisenmann, T., "Mobile Fourier-transform infrared spectroscopy monitoring of air pollution." *Appl. Optics* **1994**, *33*, 5682-5689.

[16] Qin, D., Cadet, G., "Quantitative analysis of process streams by on-line FT-IR spectrometry." *Anal. Chem*. **1997**, *69*, 1942-1945.

[17] Hart, B.K., Griffiths, P.R., "Effect of resolution on quantification in open-path Fourier transform infrared spectrometry under conditions of low detector noise. 1. Classical least squares regression." *Environ. Sci. Technol*. **2000**, *34*, 1337-1345.

[18] Hart, B.K., Berry, R.J., Griffiths, P.R., "Effect of resolution on quantification in open-path Fourier transform infrared spectrometry under conditions of low detector noise. 1. Partial least squares regression." *Environ. Sci. Technol*. **2000**, *34*, 1346-1351.

[19] Johansen, I.-R., Lines, G.T., Honne, A., Midtgaard, T., "Calibration of an FT-IR spectrometer for ambient air monitoring using PLS." *Appl. Spectrosc*. **1997**, *51*, 1540-1546.

[20] Jaakkola, P., Tate, J.D., Paakkunainen, M., Kauppinen, J., Saarinen, P., "Instrumental resolution considerations for Fourier transform infrared gas-phase spectroscopy." *Appl. Spectrosc*. **1997**, *51*, 1159-1169.

[21] International Standards Organization, ISO 3308:2000, **2000**.

[22] International Standards Organization, ISO 3402:1999, **1999**.

[23] Cooperation Centre for Scientific Research Relative to Tobacco, CORESTA Recommended Method No. 55, **2002**.

[24] Griffiths, P.R., de Haseth, J.A., *Fourier Transform Infrared Spectrometry*, John Wiley and Sons: New York, **1986**.

[25] Marten, H., Næs, T., *Multivariate Calibration*, John Wiley and Sons, Chichester, **1989**.

CHAPTER 3.

CHARACTERIZATION OF A MID-INFRARED HOLLOW WAVEGUIDE GAS CELL FOR THE ANALYSIS OF CARBON MONOXIDE AND NITRIC OXIDE

3.1. Introduction

Infrared spectroscopy is routinely used for real time analysis of gas phase analytes combining the specificity of fundamental vibrational absorptions with the ease of in situ analysis. Gas phase IR analysis does not require trapping or extraction of an analyte from its matrix. IR gas cells can be designed to accommodate the sample stream with minimal sample conditioning for either environmental[1,2] or industrial processes[3,4]. IR specificity allows for single component analysis with minimal interference from other matrix components or multi-component analysis[1]. Most IR analyzers are used for single analyte applications[5]. Non-dispersive infrared (NDIR) analyzers are commonly used for monitoring carbon monoxide (CO), carbon dioxide, methane and water in combustion effluent, such as stack gas[6]. Analysis by NDIR requires an independent analysis system for each analyte. Fourier transform infrared spectrometry has the capability to perform multi-component analysis using a single instrument[1,2,7,8]. One of the limitations of conventional FT-IR based gas measurement systems compared to NDIR analyzers is the response time. The concentration range of the different small gas phase analytes resulting from combustion can range from less than 2ppm to greater than 15000ppm. This concentration range is for small gaseous constituents of sidestream cigarette smoke (*e.g.* HCN and acrolein on the low end and CO and CO_2 on the high end). These are the targeted constituents for FT-IR analysis. Other larger molecules, like PAHs, are present

in whole sidestream smoke at the ppb range or less[9]. To perform the analyses, the FT-IR spectrometer is typically combined with a long pathlength gas cell[1,5,10]. These gas cells have an internal volume ranging from few hundreds of milliliters up to 8L. The response time of the gas analysis system is dependent upon the optical efficiency of the gas cell, defined as the ratio of the optical pathlength to the volume of the gas cell, and the flow response time of the gas cell. For long pathlength gas cells, the optical efficiency is typically less than one[10,11]. In environmental monitoring, where changes in concentration occur at a comparatively slow time scale, the system response time is usually not a limitation. However, in process monitoring and surveillance or for investigating combustion processes, timely feedback is required, rendering analysis systems with long response times less than optimal.

The capabilities and performance of commercially available FT-IR systems have significantly improved over the last decade. In addition to optical improvements (i.e. signal-to-noise levels, spectral resolution, etc.), the scanning time of FT-IR instruments has improved due to advanced control techniques.

Hollow waveguides have been evaluated for transmission of mid-infrared (MIR) radiation and successfully applied to gas sensing applications[12-23]. Worrell and Gallen[16] used a wavelength tunable CO_2 laser in conjunction with MIR hollow waveguides to measure ethylene, tricholoroethylene and 2-furaldehyde. Micheels et al.[19] coupled a coiled MIR hollow waveguide with a FT-IR to measure volatile organic compounds in the headspace of a water sample. A membrane sampling device was coupled to the hollow waveguide by de Melas, et al.[20] for the analysis of benzene, toluene and xylenes

in water. The present study characterizes a hollow waveguide coupled to a FT-IR spectrometer for the analysis of CO and NO. The compounds are by-products of many combustion processes. Hence, time resolved analysis is desirable for continuous emission monitoring and unique applications like measurement of sidestream cigarette smoke[24]. The hollow waveguide gas cell is compared to a conventional long pathlength gas cells and traditional analyzers.

3.2. Experimental Section

3.2.1. Experimental Setup

All infrared spectra were collected on a Bruker Equinox 55 FT-IR spectrometer (Bruker Optics, Inc., Billerica, MA). A 3m multi-pass gas cell (Mars, Gemini Scientific, Inc., Buena Park, CA) was placed in the sample compartment of the FT-IR and in series with a non-dispersive infrared (NDIR) CO and a chemiluminescence NO analyzers (ZRH2 and CLD400, respectively, California Analytical Instruments, Inc., Orange, CA) (Figure 3–1). The input gas stream was filtered for particulate matter with a 44mm glass fiber filter with a nominal pore size of 1μm. All connecting tubing was made from Teflon with an outer diameter of 0.25in. Gas flow was maintained with a metal bellows vacuum pump (MB-41, Senior Flexonics, Inc., Sharon, MA) and monitored with an in-line pressure and flow meter (4120A, TSI, Inc., Shoreview, MN). The inlet sample flow of the NO analyzer was kept constant at 2L/min with internal pressure regulated mass flow controllers. A needle valve was inserted at the inlet to the CO analyzer to control the total measured flow rate at 3L/min.

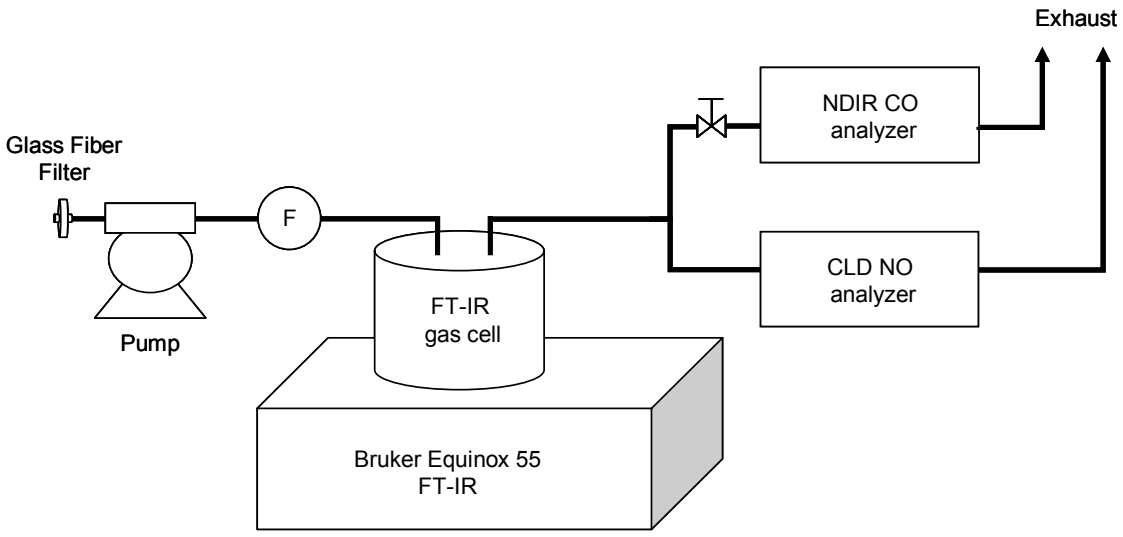

Figure 3–1. Schematic of the experimental setup for collection of infrared spectra with the 3m multi-pass gas cell.

The 2.1mm inner diameter hollow waveguide was developed by our collaborators at Tel-Aviv University (research group of N. Croitoru) and is based on a quartz structural tube coated with Ag/AgI. The HW gas cell assembly was setup on an optical train external to the FT-IR spectrometer (Figure 3–2 and Figure 3–3). The IR beam exits the side port of the FT-IR and is reflected off a 50.8mm diameter gold coated flat mirror (Janos Technology, Inc., Townshend, VT) and focused into the HW with a gold coated off-axis parabolic mirror (50.8mm diameter, 25.4mm focal length, Janos Technology, Inc., Townshend, VT).

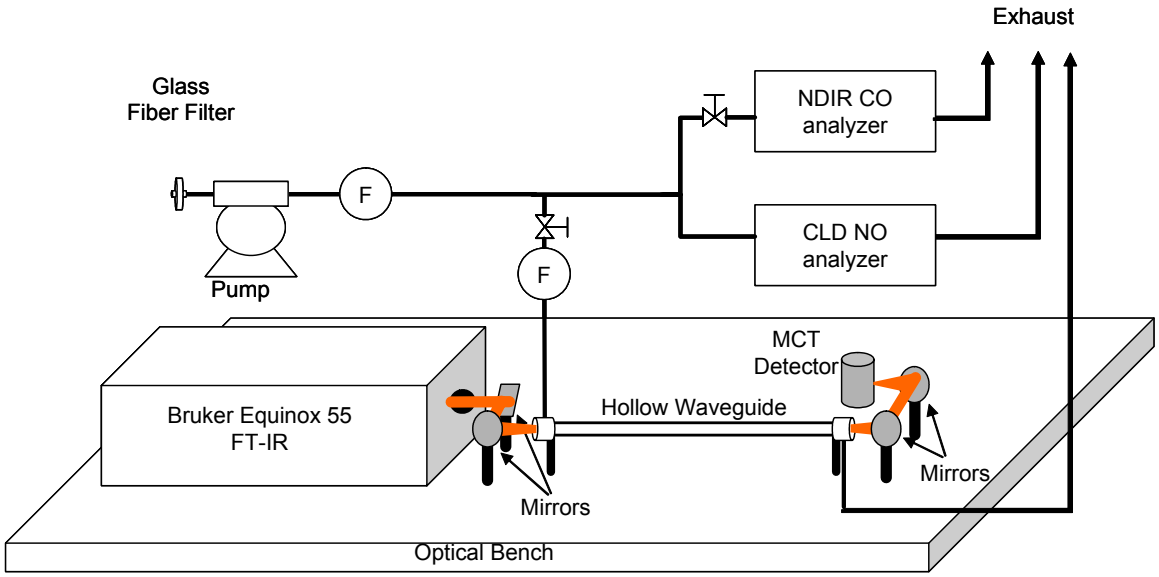

Figure 3–2. Schematic of the experimental setup for collection of infrared spectra with the hollow waveguide gas cell.

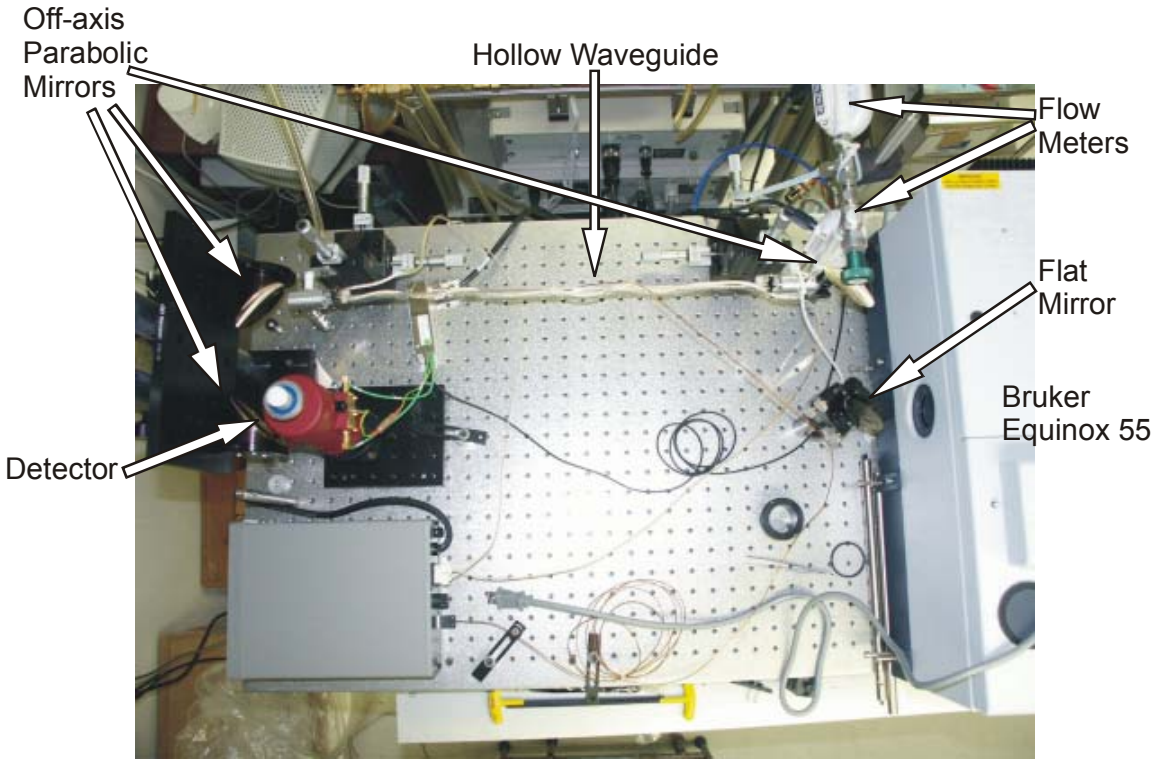

Figure 3–3. Optical setup of Bruker FT-IR with hollow waveguide gas cell.

Infrared radiation exiting the HW was collected and focused onto a liquid nitrogen cooled Hg-Cd-Te (MCT) detector (Bruker Optics, Inc., Bellerica, MA) with two gold coated off-axis parabolic mirrors (50.8mm diameter, 25.4mm focal length, Janos Technology, Inc., Townshend, VT). Similar to the 3m gas cell configuration, the gas stream passes through a glass fiber filter and the metal bellows pump. In contrast to the 3m gas cell configuration, the HW gas flow is parallel to the CO and NO analyzers. At the T-junction to the HW, a second needle valve and flow meter were inserted to regulate and

measure the HW flow rate to 0.25L/min. The total measured flow rate of the system was maintained at 3L/min.

Certified gas standards (Air Products, Inc., Theodore, AL) were used to characterize the gas cells and calibrate the FT-IR. The CO standards included 1985, 5150, 7380, and 10,100ppm. The NO standards included 49.69, 389.2, 600, 1043, and 1500ppm. All standards were introduced to the respective measurement configuration with 10L polypropylene gas bags (K.C. Automation, Inc., Richmond, VA). The bag was rinsed twice with the standard prior to filling with the sample to be introduced. After the standard was introduced, three FT-IR absorbance spectra were collected. Analog signals from the two gas analyzers and the flow meter were connected to analog-to-digital inputs of a motor controller board (ESP6000, Newport, Inc., Irvine, CA) and converted to digital values at the rate of one data point per second for storage and analysis. The data were automatically collected and stored using MATLAB (Mathworks, Inc., Natick, MA). FT-IR calibrations were developed by performing a least squares regression of concentration values versus integrated absorbance or peak height for the rotation-vibration lines of the R-branch of the absorption bands centered at 2144 and 1875cm^{-1} for CO and NO, respectively. Three different mixtures of CO and NO were generated from the 10,100ppm CO and 600ppm NO standards and instrument air using an automated gas mixing system (Environics 4000, Environics, Inc., Tolland, CT). The mixtures were introduced to the system and spectra were collected similar to the gas standards. Integrated absorbances and peak heights were calculated in OPUS software (Bruker

Optics, Inc., Bellerica, MA). All other data and statistical analyses were performed in MATLAB.

3.2.2. Determination of Flow Response Times

For the two experimental setups, the 2000ppm CO gas standard was introduced to the HW system for 30s and the multi-pass gas cell for 45s. Absorbance spectra were collected with the FT-IR in the time resolved mode at the approximate rate of one interferogram per second. The concentration readout of the NDIR CO analyzer was digitally recorded at a rate of one concentration value per second. The flow response was determined by solving the exponential decay equation for the decay constant τ,

$$A(t) = A(0)e^{(-t/\tau)} \tag{3-1}$$

where $A(t)$ and $A(0)$ are the amplitude of the signal at times t and 0, with the latter value corresponding to the beginning of the decay curve, respectively. Flow response times for the three methods of measuring CO were determined from concentration decay curves. Each curve was normalized to the magnitude of the concentration vector. The normalized curves were smoothed using the Savitzky-Golay (S-G) filter (PLS Toolbox for MATLAB, Eigenvector Research, Inc., Manson, WA) with a filter width of five. To solve Equation (3-1), the natural logarithm of the signal was taken and the slopes for each curve were computed using the S-G first derivative filter with a filter width of five. The flow response time was calculated from the slope on the decay curve with the largest coefficient of determination and the mean of time difference between the data points.

3.2.3. Determination of Sensitivity

The sensitivity was determined from the slope of the least squares regression of the concentration versus absorbance at each wavenumber or integrated absorbance of selected spectral regions and normalized to the nominal pathlength of each gas cell. All regressions were performed in MATLAB. The peak height and integrated absorbance values were calculated for each rotation-vibration line of the R-branch for each component using the absorption bands centered at 2144 and 1875cm^{-1} for CO and NO, respectively.

3.2.4. Monte Carlo Simulation.

Using the internal reflection model for hollow waveguides[25,26], a simplified Monte Carlo simulation was developed to estimated the effective pathlength of the 55cm hollow waveguide. The reflective surfaces of 25.4mm focal length parabolic mirrors and the hollow waveguide were modeled in 3-D using MATLAB. The Monte Carlo simulation launched 10^3 photons at the mirror with random start positions within a collimated beam with a diameter of 40mm. Reflections from the mirror into and through the waveguide were calculated using the law of reflection[27]. Photon path and distance were stored and the HW effective pathlength was calculated from the average distance traveled by photons through the HW in the simulation.

3.3. Results and Discussion

The flow response of a gas cell is an important parameter for sensing applications as it defines the response time of the system. The theoretical aspects of determining the time constants from the decay section of the response curve are straightforward; however, the

addition of noise complicates the calculations. The root-mean-square (RMS) noise levels of the response curves are calculated by taking the root-mean-square of the difference between the original data and data sufficiently smoothed to provide an estimate of the noiseless curve. This was evaluated empirically with the S-G filter with a filter width of 11. The resulting RMS noise is 1.8, 0.2 and 0.3% for the HW, the 3m gas cell and the CO analyzer, respectively. Using simulated data, Figure 3–4a shows four decay curves with τ = 1.23, $A(0)$ = 0.25, and 0, 0.01, 0.1 and 1% Gaussian noise added. Figure 3–4b shows the natural logarithms of those curves, which are used to regress against t to estimate the time constant. The addition of noise reduces the portion of the curve suitable for estimating τ. Without any data preprocessing, the errors in estimating τ ranged from less than 0.1% for data with little noise to >10% for 1% of added noise. Smoothing the decay curve prior to taking the logarithm reduces the error to a consistent level of approx. 2.5% at all noise levels.

The time constants of the gas cell are calculated from the concentration decay curves (Figure 3–5). The time constants for CO are 1.35, 1.23 and 12.4s for the NDIR CO analyzer, HW, and multi-pass gas cell, respectively. The response times impressively demonstrate the optical efficiency of the HW gas cell. In combination with a fast scanning FT-IR spectrometer, the HW is comparable in response time to the single component gas analyzer.

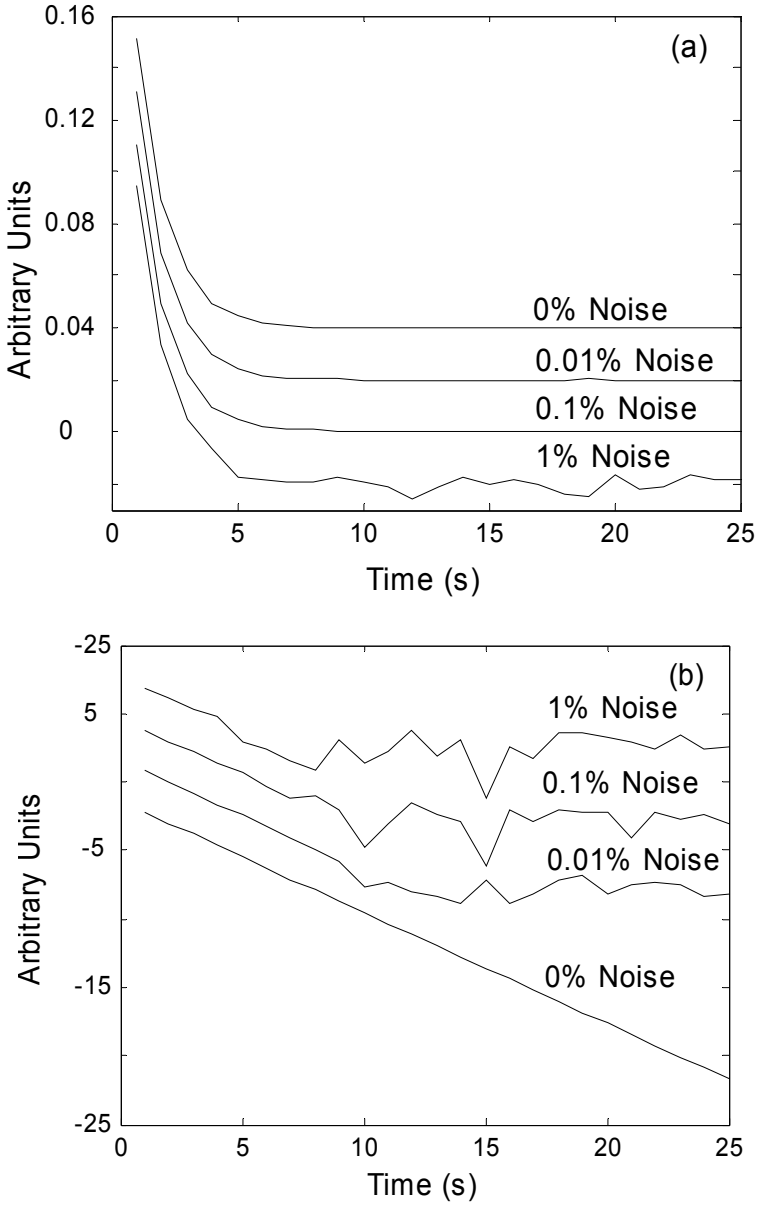

Figure 3–4. Simulated response curves with added Gaussian noise with time constant τ = 1.23 and initial amplitude A(0) = 0.25. (a) Simulated curves. (b) Natural logarithm of simulated curves. Curves are synthetically offset for clarity.

Plots of the wavenumber ($\bar{\nu}$) versus sensitivity of the HW and the 3m gas cell for CO and NO per nominal pathlength (3m for the multi-pass gas cell and 0.55m for the HW) are shown in Figure 3–6. These plots indicate that the HW reveals increased sensitivity compared to the 3m gas cell. Calibrations using the certified gas standards show increases in sensitivity ranging from -1 to 51% for CO and -6 to 59% for NO (Table 3–1).

The range in sensitivities demonstrates the ability using rotational bands to perform quantitative analysis over a wider range by using less sensitive bands at higher concentrations.

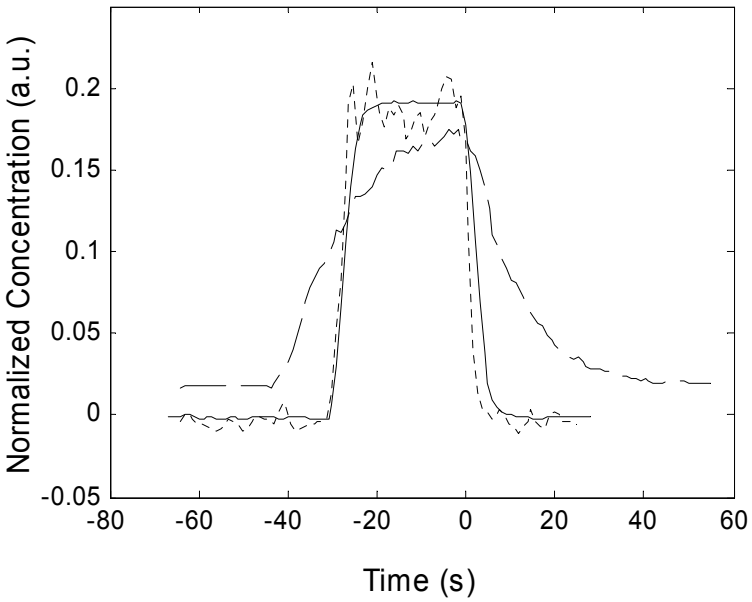

Figure 3–5. Flow response decay curves for (—) NDIR CO analyzer, (···) FT-IR with hollow waveguide gas cell and (--) FT-IR with 3m multi-pass gas cell.

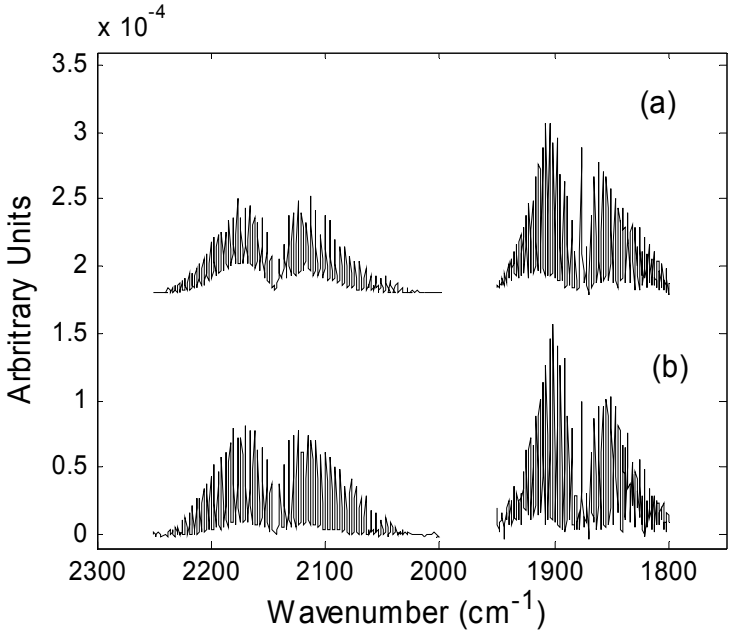

Figure 3–6. Wavenumber versus sensitivity per nominal pathlength for (a) 3m multi-pass and (b) hollow waveguide gas cell.

One rationalization for the increased sensitivity is an increase in "photon density" in the HW gas cell. In the 3m gas cell, the focal points of the folded beam are at the field mirror. This produces a series of radiation cones that fan through approximately 85% of the volume of the cylinder defined by the field mirror and the objective mirrors. Since the HW is a non-imaging optical device, nearly 100% of its volume is occupied by the IR beam. Assuming all other optical characteristics are equivalent, the HW should have an increased "photon density" of approximately 125 fold. If the movement of gas molecules and the propagation of photons in the gas are considered random events, the probability of interaction between the two should increase linearly with an increase in either domain. Therefore, a 125 fold increase in sensitivity should be realized by using the HW,

however, only an improvement factor of <1 is evident from the experimental findings. Using a 3-D Monte Carlo simulation of photons traveling through the HW with a 15° acceptance angle reveals that only approx. 3% of the photons reach the detector with this optical arrangement. Combined with the estimated attenuation for the HW of 0.5db, a 1.2 fold increase in sensitivity should be realized. This estimate, considering only optical efficiencies of the gas cells and simulated throughput of the HW is within an order of magnitude of the increase in sensitivity measured in this study.

Simulations also show that by increasing the focal length of the off-axis parabolic mirror that focuses the beam into the HW, fewer photons are rejected because of the acceptance angle. The percentage of photons reaching the detector increases to 85% when the focal length of the mirror is set to 127mm. Obviously, longer mirror focal lengths diminish the advantages of the compact design of the HW because the distance from the mirror to the HW would be nearly half the length of the HW. Still, since only 3% of the photons reach the detector in the current configuration, other optical combinations of mirrors or lenses could be applied to achieve more photons entering the HW within the 15° acceptance angle yielding a net gain in throughput.

Table 3–1. Analyte sensitivities ($ppm^{-1}m^{-1}$) based on rotational band integrated absorbances and peak intensities.

		Sensitivity					
		Integrated Absorbance			Peak Height		
Analyte	Frequency (cm^{-1})	Hollow Waveguide (x 10^{-5})	3m gas cell (x 10^{-5})	Increase (%)	Hollow Waveguide (x 10^{-5})	3m gas cell (x 10^{-5})	Increase (%)
CO	2213	3.00	2.29	30.9	2.82	1.87	50.9
	2209	3.00	2.74	9.5	2.75	2.26	21.5
	2206	3.86	3.27	18.0	3.49	2.62	33.0
	2203	4.79	4.35	10.0	4.13	2.94	40.5
	2200	4.86	4.57	6.4	4.29	3.76	14.1
	2197	5.70	5.22	9.1	5.17	4.15	24.7
	2193	5.98	5.77	3.8	5.04	4.29	17.4
	2190	6.34	5.39	17.6	6.11	4.55	34.3
	2187	6.52	5.69	14.6	6.31	4.76	32.5
	2183	7.21	6.16	17.1	7.01	5.17	35.5
	2180	8.72	8.01	8.9	8.03	5.71	40.7
	2176	8.65	8.77	-1.4	7.44	6.83	8.8
	2173	7.90	7.00	12.8	7.54	6.11	23.4
	2169	9.21	8.65	6.5	8.18	6.43	27.2
	2166	9.03	8.94	1.0	7.93	7.36	7.7
	2162	8.82	8.02	10.0	7.80	5.61	39.0
	2158	7.12	6.21	14.7	6.70	5.38	24.4
	2155	7.40	6.98	6.0	6.50	5.47	18.8
	2151	6.32	5.60	12.8	5.52	4.46	23.8
	2147	4.09	3.11	31.5	4.04	2.81	43.9
NO	1915	12.89	13.37	-3.6	1.27	1.27	-0.1
	1912	12.09	12.60	-4.1	1.42	1.44	-1.4
	1909	13.72	13.52	1.5	1.58	1.61	-1.8
	1906	14.87	14.21	4.7	1.77	1.80	-1.3
	1903	16.43	14.06	16.9	2.01	1.82	10.1
	1900	15.04	11.92	26.2	2.19	1.67	31.5
	1897	13.04	10.46	24.7	2.00	1.65	21.2
	1894	12.26	9.62	27.4	1.77	1.34	31.8
	1891	12.64	8.34	51.6	1.93	1.21	59.4
	1880	9.39	8.05	16.7	1.19	1.03	14.7
	1884	6.44	5.55	16.1	1.00	0.79	25.8
	1876	11.12	11.87	-6.3	1.52	1.57	-3.1

The regressions of the rotational bands have coefficients of determination ranging from 0.963 to 0.998 and 0.878 to 0.979 for CO and NO calibrations, respectively. Prediction results using the most robust calibrations for the gas standard mixtures are shown in Table 3–2. In general, all calibrations were not significantly different from the analyzer result. This is particularly evident in the HW results where the analyzer is measuring in parallel with the HW as opposed to in series with the 3m gas cell. The FT-IR overestimates the CO concentration. However, there is good agreement between the FT-IR results and the analyzer's for NO for both gas cells, the HW device and the 3m gas cell. Substituting an FT-IR for individual analyzers can result in a cost advantage for gas sensing applications targeting individual compounds. However, by performing multi-component analyses with one instrument, the costs for instrumentation and sample conditioning drops at a per analyte basis. Furthermore, direct comparison of different analytes is simplified by reporting results for the same physical sample. The developed HW gas cell device provides additional cost savings considering that the optical setup for the HW gas cell costs about an order of magnitude less than a commercial multi-pass gas cell.

Table 3–2. Results (ppm) from CO and NO calibrations on mixture samples.

Analyte	Gas Cell	Mixture	Analyzer	Integrated Absorbance	Peak Height
CO	HW	A	3370 ± 5	3489 ± 167	3506 ± 138
		B	1694 ± 4	1138 ± 82	1262 ± 263
		C	5156 ± 14	5479 ± 98	5459 ± 207
	3m	A	3401 ± 107	3567 ± 23	3527 ± 72
		B	1682 ± 41	1690 ± 26	1461 ± 6
		C	5108 ± 153	5126 ± 184	5239 ± 96
NO	HW	A	384.2 ± 2.1	395.7 ± 57.4	363.0 ± 24.3
		B	490.0 ± 0.8	475.2 ± 37.1	505.6 ± 35.6
		C	294.0 ± 1.1	298.3 ± 41.5	276.8 ± 65.0
	3m	A	392.9 ± 10.9	387.7 ± 3.8	387.5 ± 1.3
		B	492.9 ± 10.5	512.2 ± 4.7	510.5 ± 7.3
		C	296.1 ± 6.2	260.5 ± 2.1	252.3 ± 5.1

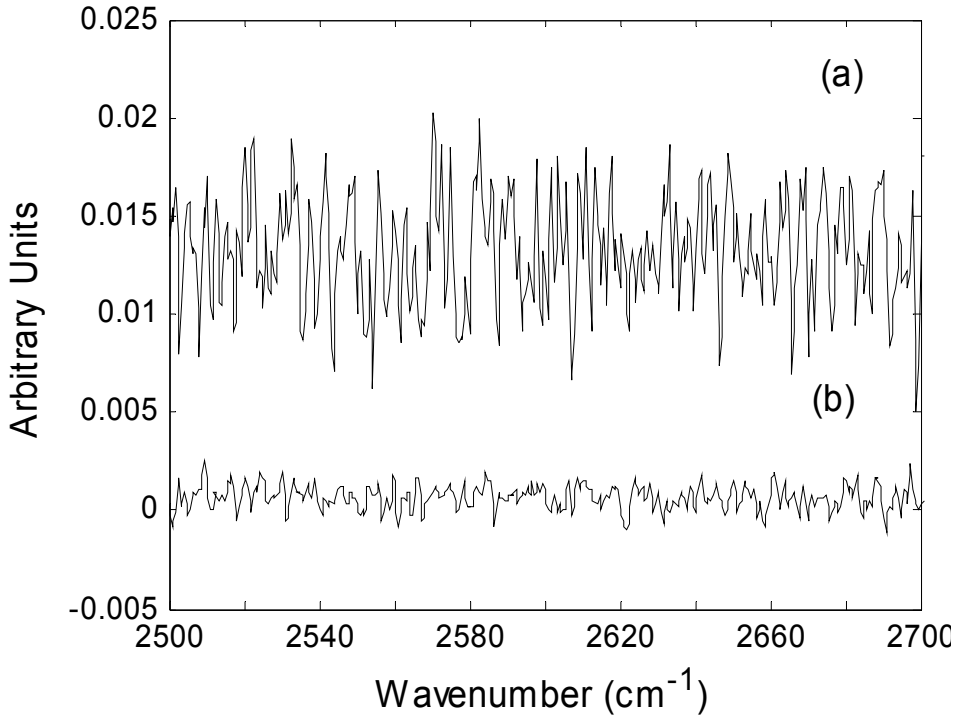

Figure 3–7. Baseline for 2000ppm CO standard in a non-absorbing region of the spectrum for (a) hollow waveguide and (b) 3m multi-pass gas cell. Curves are synthetically offset for clarity.

As evident from the measurements the HW results show a tendency to be less reproducible. This is due to the mechanical instability of the HW experimental setup used in the present device configuration. Since the waveguide is not secured to rigid mounts across its length, vibrations due to pressure pulses from the gas sampling pump can resonate along the waveguide resulting in mechanical movement of the device. The signal to noise level for the CO absorption band for the 3m gas cell is 1.4 times greater than for the HW. The RMS baseline noise for the HW in a non-absorbing region of the mid-infrared is 3.8 times that of the 3m gas cell (Figure 3–7). Despite the increase in noise, the HW still provides comparable performance to the 3m multi-pass gas cell. It is expected that improvements in performance can be obtained through modifications of the experimental setup. For example, if by securing the length of the HW to the optical table and minimizing the pressure pulses reduces the noise in the baseline to the level measured in the 3m gas cell, a four fold improvement of the signal-to-noise level could be obtained.

3.4. Conclusions

A hollow waveguide gas cell with a length of 550mm and an inner diameter of 2.1mm has been shown to provide comparable performance to a 3m multi-pass gas cell for FT-IR analysis of CO and NO. The HW response time is one-tenth that of the larger volume multi-pass gas cell and slightly faster than a single analyte gas analyzer, which is an important characteristic for time resolved sensing applications. Furthermore, it has been demonstrated that the HW provides up to approx. 60% greater sensitivity on a per meter optical path basis than the multi-pass gas cell of the analytes investigated due to increased

optical efficiency maximizing the light concentration within the gaseous sample volume. Despite the multi-pass gas cell having 40% greater signal-to-noise, there are opportunities to reduce the RMS noise in the HW experimental set-up and improve the signal-to-noise metric. These characteristics will lead to the development of more compact instrumentation for gas sensing applications without sacrificing the quantitative performance. Multi-component analyses for both gas cells were not significantly different from the results obtained by individual analyzers, despite the increased noise levels of the HW device. Finally, substantial cost savings for gas sensing applications are attainable with the developed setup by maximizing the number of analytes measured within a given sample.

3.5. References

[1] Elser, M.B., Griffith, D.W.T., Wilson, S.R., Steele, L.P., *Anal. Chem*. **2000**,*72*, 206.

[2] Johansen, I-R., Lines, G.T., Honne, A., Midtgaard, T., "Calibration of an FT-IR spectrometer for ambient air monitoring using PLS." *Appl. Spectrosc*. **1997**, *51*, 1540.

[3] Martin, D.M., Medvecz, P.J., Nichols, K.M., "Application of FT-IR absorption spectroscopy for *in situ* gas concentration and temperature measurements in laboratory- and pilot-scale combustion environments." *Appl. Spectrosc*. **1993**, *47*, 1898.

[4] Qin, D., Cadet, G., "Quantitative analysis of process streams by on-line FT-IR spectrometry." *Anal. Chem*. **1997**, *69*, 1942.

[5] Woebkenburg, M.L., "Portable infrared analyzers.: *Appl. Occup. Environ. Hyg*. **1999**, *14*, 510.

[6] Herget, W.F., Jahnke, J.A., Burch, D.E., Gryvnak, D.A., "Infrared gas-filter correlation instrument for *in situ* measurement of gaseous pollutant concentrations." *Appl. Opt*. **1976**, *15*, 1222.

[7] Welch, G.M., Herman, B.E., *Adv. Instrum. Contrl*. **1995**, *50*, 65.

[8] Gaudex, M.T., Pitsch, H., Florestan, J., Boulou, J.C., "Quantitative analysis of small volumes of gas mixtures by Fourier transform infrared spectroscopy." *Analyst* **1994**, *119*, 665.

[9] Baker, R.R. in *Tobacco Production, Chemistry and Technology*, Davis, D.L., Nielson, M.T., Eds.; Blackwell Science, London, **1999**.

[10] Hanst, P.L., "Air pollution measurement by Fourier transform spectroscopy." *Appl. Opt*. **1978**, *17*, 1360.

[11] White, J.U., *J. Opt. Soc. Am*. **1942**, *32*, 285.

[12] Worrell, C.A., Giles, I.P., Adatia, N.A., "Remote gas sensing with mid-infrared hollow waveguide." *Electron. Lett*. **1992**, *28*, 615.

[13] Saggese, S.J., Harrington, J.A., Sigel, Jr., G.H., *Proc. SPIE-Int. Soc. Opt. Eng*. **1991**, *1368*, 2.

[14] Saito, S., Sato, S., Miyagi, M., *Proc. SPIE-Int. Soc. Opt. Eng*. **1992**, *1796*, 231.

[15] Saito, S., Sato, S., Miyagi, M., "Infrared hollow waveguides for capillary flow cells." *Appl. Spectrosc*. **1993**, *47*, 1665.

[16] Kozodoy, R.L., Micheels, R.H., Harrington, J.A., "Small-bore hollow waveguide infrared absorption cells for gas sensing." *Appl. Spectrosc*. **1996**, *50*, 415.

[17] Worrell, C.A., Gallen, N.A., "Trace-level detection of gases and vapours with mid-infrared hollow waveguides." *J. Phys. D: Appl. Phys*. **1997**, *30*, 1984.

[18] Haan, D.J., Gibson, D.J., Rabii, C.D., Harrington, J.A., "Coiled hollow waveguides for gas sensing." *Proc. SPIE-Int. Soc. Opt. Eng*. **1998**, *3262*, 125.

[19] Micheels, R.H., Richardson, K., Haan, D.J., Harrington, J.A., "FT-IR based instrument employing a coiled hollow waveguide cell for rapid field analysis of volatile organic compounds." *Proc. SPIE-Int. Soc. Opt. Eng.* **1999**, *3540*, 64.

[20] de Melas, F., Pustogov, V.V., Croitoru, N., Mizaikoff, B., "Development and optimization of a mid-infrared hollow waveguide gas sensor combined with a supported capillary membrane sampler." *Appl. Spectrosc*. **2003**, *57*, 600.

[21] Pogodina, O. A., Pustogov, V. V., de Melas, F., Haberhauer-Troyer, C., Rosenberg, E., Puxbaum, H., Inberg, A., Croitoru, N., Mizaikoff, B., *Anal. Chem*., **2004**, *76*, 464.

[22] de Melas, F., Pustogov, V. V., Wolcott, D. K., Olson, D. C., Inberg, A., Croitoru, N., Mizaikoff, B., *Internat. J. Env. Analyt. Chem*., **2003**, *83*, 573.

[23] Charlton, C., Mizaikoff, B., de Melas, F., Inberg, A., Croitoru, N., *IEE Optoelectronics*, **2003**, *150*, 306.

[24] Thompson, B.T., Mizaikoff, B., "Real-time FT-IR analysis of carbon monoxide and nitric oxide in sidestream cigarette smoke." submitted to *Appl Spectrosc*..

[25] Alaluf, M., Dror, J., Dahan, R., Croitoru, N.J., "Plastic hollow fibers as a selective infrared radiation transmitting medium." *J. Appl. Phys*. **1992**, *72*, 3878.

[26] Wolfbeis, O.S., "Capillary waveguide sensors." *Trends Anal. Chem*. **1996**, *15*, 225.

[27] Born, M., Wolf, E., *Principles of Optics* , 6th ed., Pergamon Press, New York, **1980**.

CHAPTER 4.

APPLICATION OF A MID-INFRARED HOLLOW WAVEGUIDE GAS CELL FOR REAL-TIME FT-IR ANALYSIS OF CARBON MONOXIDE IN SIDESTREAM CIGARETTE SMOKE

4.1. Introduction

Various types of infrared spectroscopy have been applied to the analysis of cigarette smoke[1-9]. Cole and Martin[8] measured the emission of several gas phase components of sidestream cigarette smoke, i.e. smoke emitted from the lit end of the cigarette, by FT-IR techniques with univariate calibrations. In Chapter 2, application of PLS to the analysis of sidestream cigarette smoke achieving comparable performance at a resolution of $8cm^{-1}$ was demonstrated. However, both studies relied on long pathlength gas cells with volumes ranging from few hundreds of milliliters to 8L. Consequently, the FT-IR results lacked the temporal resolution attainable by the non-dispersive infrared (NDIR) analyzer for carbon monoxide (CO). Shi, et al.[9] used a quantum cascade laser (QCL) and a 0.3L gas cell under constant vacuum to achieve the desired time resolution for monitoring inter-puff concentration profiles for several gas phase sidestream smoke analytes. The QCL spectrometer was successful in monitoring sidestream smoke, but required independent carbon dioxide measurements obtaining emission ratios and emitted amounts for the gas phase analytes. Clearly, the increased temporal resolution aides in understanding the combustion gas formation during and between puffs. For sidestream smoke, combustion product emission between products generated from the puff and from smoldering (inter-puff period) needs to be resolved on the basis of a single cigarette.

These results are of increasing importance as new types of cigarette papers are used to meet self-extinguishing standards for "fire safe" cigarettes.

In order to achieve the demanded combination of temporal resolution and accurate yields, FT-IR spectroscopy appears a suitable method collecting data with a fast scanning spectrometer in combination with robust calibrations for quantitative results. Cigarette smoke is known to contain thousands of compounds and the amounts of vapor phase components in the sidestream smoke are emitted at levels two to ten times the mainstream amounts[10,11]. In order to utilize rotation-vibration lines as reported in previous studies[8,9], FT-IR spectra would have to be collected at a spectral resolution of $1cm^{-1}$ or better. Improved temporal resolution of the sidestream concentration profile could only be attained at an equivalent signal-to-noise level by collecting spectra at lower resolution. However, at lower resolution ($>1cm^{-1}$) the rotation-vibration lines would no longer be spectroscopically resolved for quantitation.

Multivariate regression techniques, such as partial least squares (PLS), could be used for accurate quantitation in the presence of interfering compounds at unknown levels. Multivariate calibrations are more robust to interferences and in dynamic systems because they use more spectral information for quantitation. FT-IR spectroscopy and PLS have been applied to multi-component analysis of combustion gases[12-15], process streams[16] and environmental pollutants[17-19]. Hart, et al.[17,18] studied the effect of spectral resolution and regression techniques on the performance of open path FT-IR for monitoring environmental pollutants. Qin and Cadet[16] evaluated the combination of low resolution FT-IR spectra and PLS to characterize effluent from a high temperature

process containing organic compounds. Jaakkola, et al.[20] studied the relationship between FT-IR instrument resolution and analytical figures of merit for the analysis of hydrocarbon mixtures. Each of these previous studies achieved successful quantitation at lower spectral resolution.

The present study investigates the application of a HW gas cell in combination with fast scanning FT-IR spectroscopy for monitoring of carbon monoxide in sidestream cigarette smoke and compares the performance of the HW gas cell to a 3m multi-pass gas cell and an independent analytical technique.

4.2. Experimental Section

The equipment, materials and procedures for generating sidestream smoke under controlled conditions have been described in detail, in Chapter 2. Briefly, data were collected with a Bruker Equinox 55 FT-IR (Bruker Optics, Inc., Bellerica, MA) equipped with either a 3m multi-pass gas cell (Gemini Scientific Instruments, Buena Park, CA) with the NDIR CO analyzer in series (Figure 4–1) or with a 0.55m mid-infrared hollow waveguide with the NDIR CO analyzer in parallel (Figure 4–2). The 3m gas cell has an internal volume of 375mL and was heated to 50°C. The 2.1mm inner diameter hollow waveguide was developed by our collaborators at Tel-Aviv University and is based on a quartz structural tube coated with Ag/AgI. The HW gas cell assembly was setup on an optical train external to the FT-IR spectrometer (Figure 4–2). The FT-IR scan parameters were 64 scans at 100kHz and a spectral resolution of $1cm^{-1}$, yielding a scan time of 21s. Data were collected in the time resolved mode where interferograms are first collected and stored in the memory during the smoke run and processed at the end of the smoke

run. The number of scans collected during a smoke run ranged from 80 to 122 depending on the amount of time needed to smoke three cigarettes in succession on the smoke machine. Data for three replicate runs of five different cigarettes, three reference cigarettes (Kentucky reference cigarette 1R4F (KY1R4F), CORESTA monitor 3 (CM3), and Industry Monitor 16 (IM16)) and two commercial cigarettes (A and B), representing the expected range of total particulate matter (TPM) delivery and ventilation of products to be tested were collected.

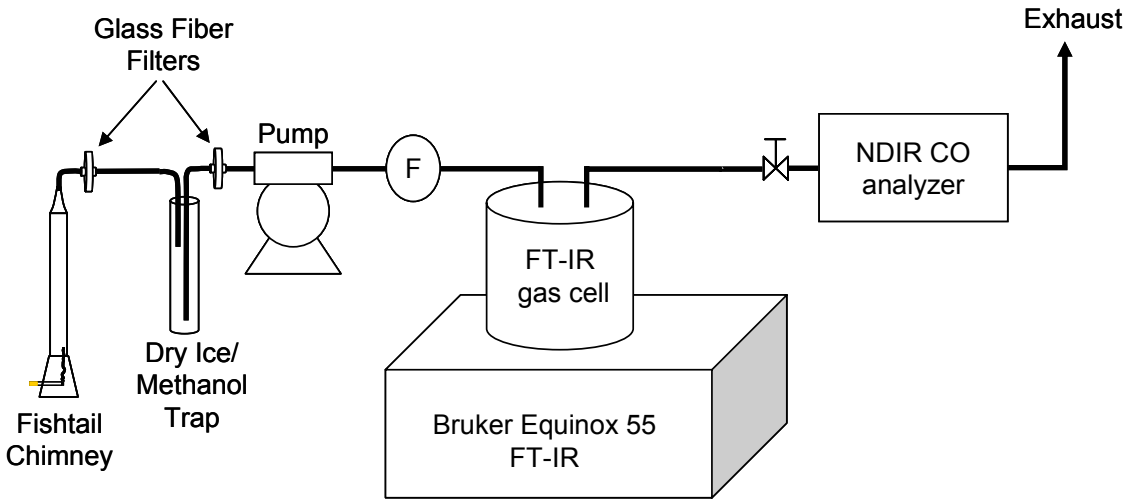

Figure 4–1. Schematic layout of experimental set-up for real time analysis of sidestream cigarette smoke using FT-IR spectroscopy with a multipath gas cell.

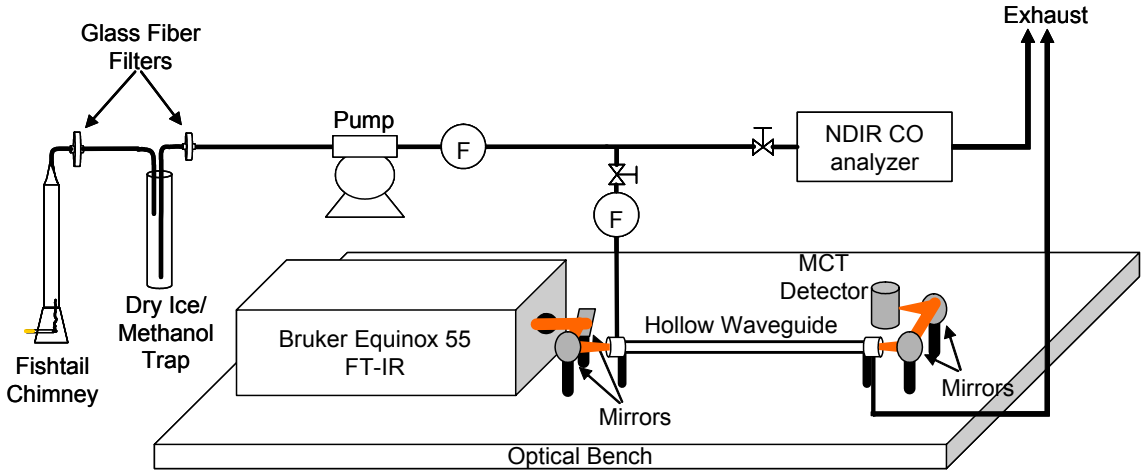

Figure 4–2. Schematic layout of experimental set-up for real time analysis of sidestream cigarette smoke using FT-IR spectroscopy with a hollow waveguide gas cell.

A non-dispersive infrared (NDIR) CO gas analyzer (ZRH2, California Analytical Instruments, Inc., Orange, CA) was located downstream of the 3m gas cell or in parallel to the HW gas flow. Analog signals from the gas analyzer and the flow meters were connected to analog-to-digital inputs of a motor controller board (ESP6000, Newport, Inc., Irvine, CA) and converted to digital values at the rate of one data point per second for storage and analysis. The data were automatically collected and stored using MATLAB (Mathworks, Inc., Natick, MA). Gas analyzer data were converted from recorded volts to concentration values and finally to the component amount in MATLAB. The amounts per second were summed up over the smoke run and divided by the number of cigarettes smoked to report yields per cigarette.

4.2.1. Calibration Methodology

The FT-IR was calibrated using a series of gas standards generated by a gas mixing system (Environics 4000, Environics, Inc., Tolland, CT). A certified calibration gases, 1.01% CO with nitrogen balance (Air Products, Inc., Theodore, AL), was applied, which had a certificate of analysis of 2% accuracy. Eight standards ranging from 500 to 4000ppm were prepared. Each standard generated from the gas mixing system was collected in a 10L polypropylene gas bag (KC Automation, Inc., Richmond, VA). The gas bag was attached to the first filter pad holder and the calibration gas was introduced to the system in a similar manner as the sidestream smoke. The system was purged for 60s with the gas standard prior to FT-IR data collection. Three FT-IR spectra versus a nitrogen background and the analog output of the analyzer were collected for each standard. The analyzer output was converted to concentration and smoothed using a moving average digital filter with a window size equal to the interferogram scan time. Each calibration gas spectrum was assigned the CO concentration from the smoothed data of the analyzer at the data collection end time of the FT-IR.

Calibrations and subsequent data analysis were performed within three different spectral regions and by both univariate and multivariate calibration approaches, as described in Chapter 2. Two different regressions were used in this study. A univariate regression performed a least squares regression of concentration against the integrated absorbance within the evaluated spectral region. The multivariate regression used a PLS approach. The regression programs were part of the OPUS FT-IR software (Bruker Optics, Inc., Bellerica, MA).

4.2.2. Calculation of Yield and Data Analysis

The FT-IR data were analyzed for each smoke run. Sidestream CO yields were calculated as described in Chapter 2. Averages and standard deviations of the yields for each cigarette type from the analyzers and all FT-IR calibrations and gas cells were calculated. Because the analyzer and FT-IR data were collected concurrently, two statistics were used to directly compare FT-IR calibration results to analyzer results. The t-statistic was calculated from the difference of each pair of results,

$$t_i = \left|\overline{d}_i\right| m^{1/2} s_{d_i}^{-1} \tag{4-1}$$

where $\overline{d}_i$ and s_{d_i} were the mean and standard deviation of paired data of component *i*, and *m* was the number of smoke runs. The root mean squared error of prediction (RMSEP) was calculated using

$$RMSEP = \left(m^{-1} \sum_{j=1}^{m} \left(Y_{i,j,\text{FT-IR}} - Y_{i,j,\text{analyzer}} \right)^2 \right)^{1/2} \tag{4-2}$$

where *m* was the number of smoke runs and $Y_{i,j,\text{FT-IR}}$ and $Y_{i,j,\text{analyzer}}$ were the yields of component *i* per cigarette for smoke run *j* for a FT-IR calibration and the analyzer, respectively.

4.3. Results and Discussion

Figure 4–3 shows the real time traces of the CO concentration for a smoke runs of the IM16 using the 3m multi-pass and HW gas cells. Evident in both plots are the breaks between the three cigarettes smoked during the run and the slight but noticeable increases in concentration due to puffing. The improvement temporal resolution due to the HW is

evident with the narrowing of the increases in concentration due to puffing in both the analyzer and FT-IR traces.

The FT-IR trace provides less temporal resolution than the analyzer trace due to a 21s sampling interval versus 1s provided by the analyzer. The scan time of the FT-IR at 21s is a limitation due to the volume and flow response time of the 3m gas cell and is applied to the HW data collection to maintain comparable instrument signal-to-noise levels. Even so, the HW shows better temporal responsiveness due to its cell flow response being one-tenth that of the 3m gas cell (Chapter 3). Being able to collect analyzer data in parallel to the HW system shows the true variability in concentration during the smoke run. The variability during the smoke run demonstrates the need to perform real time analyses for gas phase smoke constituents with sufficient temporal resolution in order to measure inter-puff concentrations. Figure 4–4 demonstrates the ability of the FT-IR to perform with the required temporal resolution. The figure shows CO concentration traces for one KY1R4F cigarette with both analyzer and FT-IR data collected at intervals of one second (collection of 16 scans at a resolution of 8cm-1). Despite the increased noise due to the reduced scan time, the responsiveness of the HW system to changes in CO concentration are comparable to that of the NDIR analyzer.

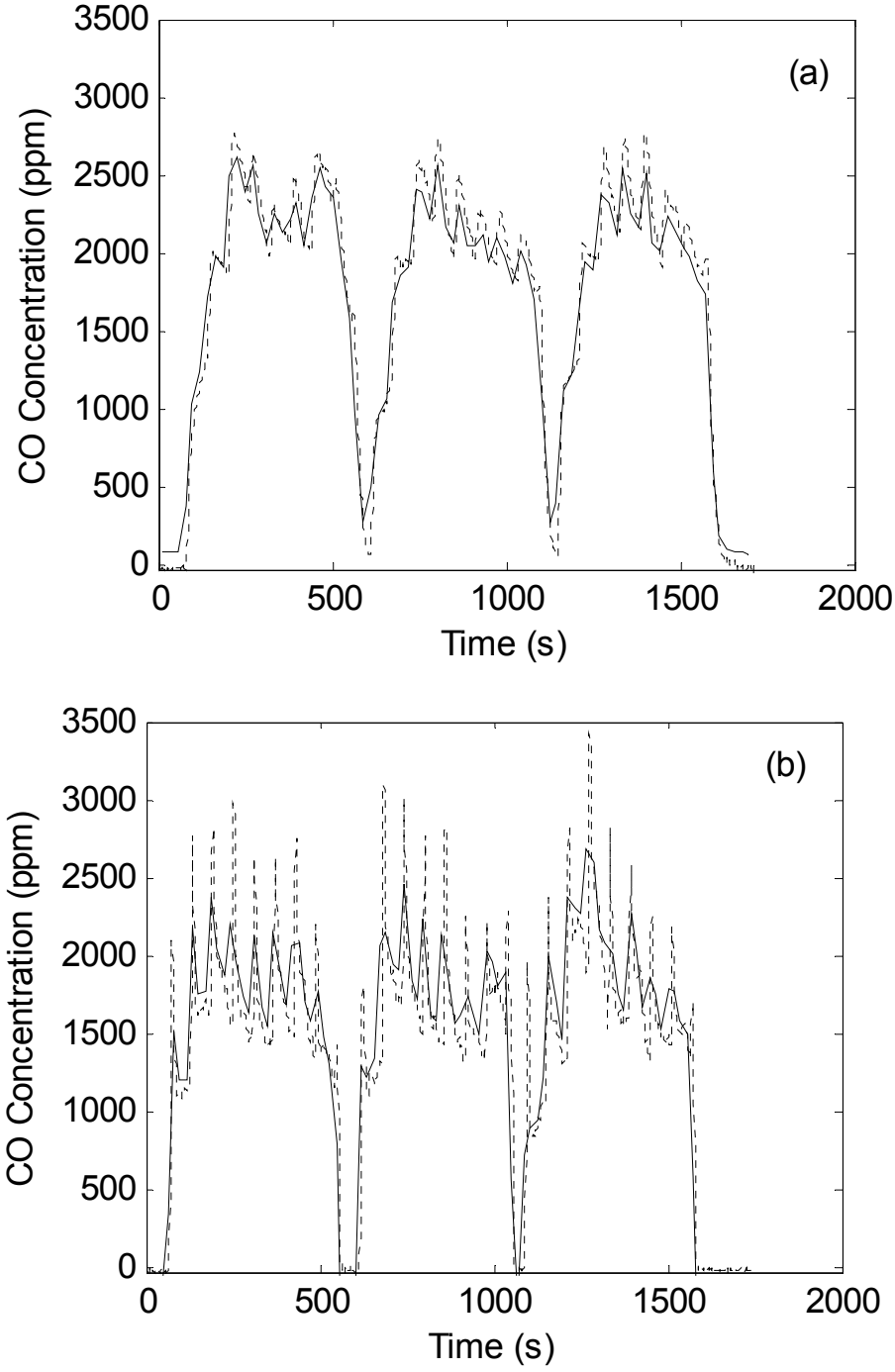

Figure 4–3. Carbon monoxide concentration traces for IM16 reference cigarette smoke runs (a) with 3m and (b) HW gas cells (··· NDIR analyzer, — FT-IR using PLS calibrations).

The cigarette results are reported as mass of CO emitted per cigarette (Table 4–1). The ranks used for the PLS calibrations are selected based on lowest RMSEP of the yields and not concentration results for individual spectra. The RMSEP is calculated from the difference in yields between the FT-IR and the analyzer results for the individual smoke runs. Because the 3m and HW gas cells required different optical setups, there are separate analyzer results for each gas cell. The difference in the analyzer results for the same cigarette collected at different times demonstrates the inherent variability of the sample and the need to measure the same physical sample when comparing methodologies. Using analyses from independent sample runs to estimate cigarette yields [9] can increase the inaccuracy of the calculated values.

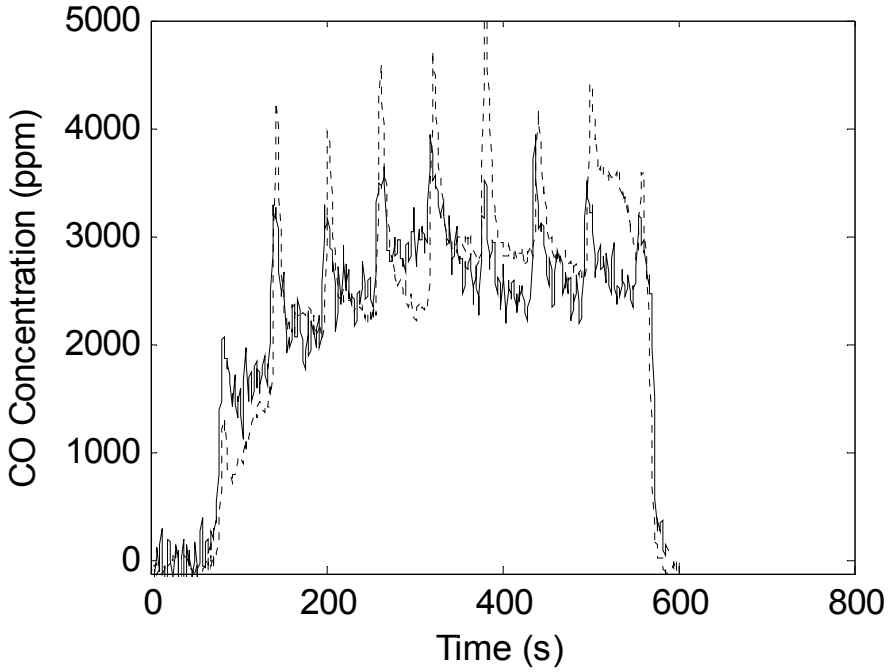

Figure 4–4. Carbon monoxide concentration traces for Kentucky reference cigarette 1R4F smoke run with FT-IR data collection at 1 s/spectrum and using the HW gas cell (··· NDIR analyzer, — FT-IR).

Table 4–1. Results for CO (mg/cig.) for five different cigarettes tested.

Cell	Regression	Spectral Range	Cigarette				
			CM3	KY1R4F	IM16	A	B
3m	Analyzer		62.4 ± 4.3	66.6 ± 3.2	57.6 ± 4.6	49.5 ± 4.0	57.9 ± 5.6
	Univariate	Rotational	61.7 ± 3.9	65.4 ± 2.3	57.6 ± 2.4	49.0 ± 3.5	57.1 ± 4.9
		R-Branch	58.5 ± 3.3	61.4 ± 3.3	53.8 ± 3.4	45.8 ± 3.8	54.1 ± 4.6
		Entire	58.0 ± 3.0	60.4 ± 3.5	53.0 ± 3.4	45.3 ± 3.7	53.5 ± 5.2
	PLS	Rotational	64.2 ± 8.2	74.5 ± 4.1	63.8 ± 1.2	52.8 ± 9.5	62.1 ± 6.9
		R-Branch	57.4 ± 2.1	60.4 ± 2.9	52.7 ± 1.6	44.8 ± 2.0	53.8 ± 4.8
		Entire	57.4 ± 2.4	61.2 ± 3.2	53.2 ± 1.5	45.1 ± 2.1	54.4 ± 5.5
HW	Analyzer		52.0 ± 1.2	51.4 ± 1.3	46.7 ± 3.3	40.9 ± 3.6	46.8 ± 2.9
	Univariate	Rotational	59.4 ± 13.6	60.9 ± 9.1	52.0 ± 16.1	47.7 ± 18.0	57.4 ± 28.1
		R-Branch	52.4 ± 6.5	49.6 ± 2.0	47.0 ± 4.7	39.8 ± 6.8	43.5 ± 6.4
		Entire	50.9 ± 9.2	49.9 ± 5.3	45.9 ± 2.5	40.3 ± 9.6	44.6 ± 5.7
	PLS	Rotational	53.4 ± 2.8	55.4 ± 1.2	49.3 ± 5.8	41.0 ± 7.3	45.7 ± 8.7
		R-Branch	54.8 ± 3.4	52.6 ± 1.2	48.8 ± 1.7	41.6 ± 5.0	45.5 ± 7.1
		Entire	53.9 ± 1.3	52.6 ± 1.7	48.9 ± 1.1	42.0 ± 4.9	45.8 ± 5.9

The obtained results show that the FT-IR calibrations using the 3m gas cell underestimate the CO yield in comparison to the yields obtained by NDIR measurements. There are instances, like the 3m gas cell using PLS on a rotation-vibration line, where the average of a one or two cigarettes are greater than the reported average for the analyzers. Comparing the mean difference for the paired data shows all 3m gas cell calibrations underestimated the analyzer results. All HW PLS calibrations overestimated the analyzer yield by approx. 1mg/cig. The HW PLS calibrations using more spectral information reduced the bias by 75% over their 3m gas cell counterparts. In general, the biases were approx. 7 and 2% for the 3m and HW gas cells, respectively. This demonstrates another significant gain by using the HW over of the 3m multi-pass gas cell.

The relative uncertainties in the CO results are on average approx. 7% for 3m calibrations. The HW relative uncertainty was about double that of the 3m gas cell. For the spectral ranges and calibration types studied, the relative uncertainties range from 0.7 to 11% and 2.2 to 49% for the 3m and HW gas cells, respectively. The increase in uncertainty for the HW is related to the decrease in signal-to-noise level of the HW relative to the 3m gas cell, as shown in Chapter 3. Another source of variability is the cigarette itself. The relative standard deviation of the analyzer results across cigarettes and runs is approximately the same as the relative uncertainty of the analyses, indicating that the analysis is contributing minimal error to the estimated yields.

The t-statistic and RMSEP results are shown in Table 4–2. In general for the t-statistic, the PLS calibrations for the 3m gas cell performed better than univariate calibrations for the three spectral ranges studied, whereas all the HW calibrations gave comparable

performance. For both the 3m and HW gas cells, calibrations showing no significant difference from the NDIR analyzer are available. Except for calibrations using the rotation-vibration line spectral region, the HW shows improvement in RMSEP over the 3m gas cell. Also, the PLS calibrations show improvement in RMSEP over univariate calibrations using the same spectral region except for the 3m gas cell using the rotation-vibration line spectral region. Further improvements in HW performance can be expected as the optical configuration is optimized and the mechanical set-up of the HW is improved. This combination should increase the signal-to-noise ratio by at least a factor of four.

Table 4–2. Statistical results for CO for the different gas cells and calibrations tested. With α=0.05 and 14 degrees of freedom the two tailed critical t-value is 2.145.

Cell	Regression	Spectral Range	t-statistic	RMSEP (mg/cig.)
3m	Univariate	Rotational	3.58	0.88
		R-Branch	23.8	4.13
		Entire	21.9	4.81
	PLS	Rotational	0.45	1.32
		R-Branch	5.90	3.39
		Entire	6.08	3.69
HW	Univariate	Rotational	4.64	10.17
		R-Branch	1.80	2.57
		Entire	2.14	2.56
	PLS	Rotational	2.10	2.87
		R-Branch	2.25	2.10
		Entire	2.82	1.75

4.4. Conclusions

Improvements to the application of FT-IR spectroscopy for the measurement of CO in sidestream cigarette smoke have been demonstrated. Application of the HW gas cell in the experimental set-up yielded enhanced temporal resolution of the smoke run. The obtained data were able to discriminate smoke generated between puffs from at the puff. The lower volume gas cell reduces the flow response time of the cell, which complements the faster scanning FT-IR system to produce data temporarily comparable to the NDIR analyzer. In comparison to the 3m gas cell, the FT-IR with HW gas cell and PLS calibrations reduced the bias between the FT-IR and NDIR analyzer by 75%. Although, relative uncertainties for the HW results were about double those of the 3m gas cell, the HW calibrations, in general, produced lower RMSEP and reduced them by about a third for calibrations utilizing a larger spectral range. Additionally, calibrations were developed for each gas cell that showed no significant difference from the independent analytical technique. Given this performance, application of the HW gas cell provided significant improvements to the analysis of gas phase sidestream cigarette smoke.

4.5. References

[1] Williams, T.B.; Belk, C.W. "An infrared method for the determination of carbon monoxide and carbon dioxide levels in cigarette smoke." *Beitr. zur Tabakforsch.*, **1972**, *6*, 210-215.

[2] Vilcins, G.; Lephardt, J.O., "Fourier transform infrared spectroscopy applied to cigarette smoke analysis." *Recent.. Adv. Tob. Sci.*, **1976**, *1*, 123-146.

[3] Maddox, W.L.; Mamantov, G. , "Analysis of cigarette smoke by Fourier transform infrared spectrometry." *Anal. Chem.*, **1977**, *49*, 331-336.

[4] Williams, T.B. , "The determination of nitric oxide in gas phase cigarette smoke by non-dispersive infrared analysis. " *Beitr. zur Tabakforsch. Intl.*, **1980**, *10*, 91-99.

[5] Cueto, R.; Church, D.F.; Pryor, W.A. , "Quantitative Fourier transform infrared analysis of gas phase cigarette smoke and other gas mixtures." *Anal. Lett.*, **1989**, *22*, 751-763.

[6] Cueto, R. , "Quantitative FTIR analysis of gas phase cigarette smoke." PhD Dissertation, Louisiana State University and Agricultural and Mechanical College, **1990**.

[7] Parrish, M.E.; Lyons-Hart, J.L.; Shafer, K.H. , "Puff-by-puff and intrapuff analysis of cigarette smoke using infrared spectroscopy." *Vibrational. Spec*., **2001**, *27*, 29-42.

[8] Cole, S.K.; Martin, P. , "Determination of gas-phase sidestream cigarette smoke components using Fourier transform infrared spectrometry." *Analyst*, **1996**, *121*, 495-500.

[9] Shi, Q.; Nelson, D.D.; McManus, J.B.; Zahniser, M.S.; Parrish, M.E.; Baren, R.E.; Shafer, K.H.; Harward, C.N. , "Quantum cascade infrared laser spectroscopy for real-time cigarette smoke analysis." *Anal. Chem.*, **2003**, *75*, 5180-5190.

[10] Dube, M. F.; Green, C.R. *Recent Advances in Tobacco Science* **1982**, *8*, 42-102.

[11] Baker, R.R. In *Tobacco Production, Chemistry and Technology*, Davis, D.L., Nielson, M.T., Eds.; Blackwell Science, London, **1999**.

[12] Martin, D.M.; Medvecz, P.J.; Nichols, K.M. , "Application of FT-IR absorption spectroscopy for *in situ* gas concentration and temperature measurements in laboratory- and pilot-scale combustion environments." *Appl. Spectroc.* **1993**, *47*, 1898-1906.

[13] Speitel, L.C. , "Fourier transform infrared analysis of combustion gases." *J. Fire Sci.* **2002**, *20*, 349-371.

[14] Blackwood, T.R., "An evaluation of flare combustion efficiency using open path Fourier transform infrared technology." *J. Air Waste Manage. Assoc.* **2000**, *50*, 1714-1722.

[15] Haus, R.; Schafer, K.; Bautzer, W.; Heland, J.; Mosebach, H.; Bittner, H.; Eisenmann, T., "Mobile Fourier-transform infrared spectroscopy monitoring of air pollution." *Appl. Optics* **1994**, *33*, 5682-5689.

[16] Qin, D.; Cadet, G., "Quantitative analysis of process streams by on-line FT-IR spectrometry." *Anal. Chem.* **1997**, *69*, 1942-1945.

[17] Hart, B.K.; Griffiths, P.R., "Effect of resolution on quantification in open-path Fourier transform infrared spectrometry under conditions of low detector noise. 1. Classical least squares regression." *Environ. Sci. Technol.* **2000**, *34*, 1337-1345.

[18] Hart, B.K.; Berry, R.J.; Griffiths, P.R., "Effect of resolution on quantification in open-path Fourier transform infrared spectrometry under conditions of low detector noise. 1. Partial least squares regression." *Environ. Sci. Technol.* **2000**, *34*, 1346-1351.

[19] Johansen, I.-R.; Lines, G.T.; Honne, A.; Midtgaard, T., "Calibration of an FT-IR spectrometer for ambient air monitoring using PLS." *Appl. Spectrosc.* **1997**, *51*, 1540-1546.

[20] Jaakkola, P.; Tate, J.D.; Paakkunainen, M.; Kauppinen, J.; Saarinen, P. , "Instrumental resolution considerations for Fourier transform infrared gas-phase spectroscopy." *Appl. Spectrosc.* **1997**, *51*, 1159-1169.

CHAPTER 5.

SIMULATION OF OPTICAL CONFIGURATION

5.1. Introduction

The benefits of optical simulation address some critical issues in sensor development. Primarily, some optical components, particularly the novel components, like hollow waveguides and concentrators, are not commercially available and have to be fabricated in the research lab. Alternatively, there may be a commercial source to manufacture the component to specifications. In either case, each iteration during sensor development has associated material and time costs.

Optical systems can be mathematically modeled using the laws or reflection and refraction. Additionally, spectral responses can be estimated for the materials used in the simulation. The application of optical simulations to sensor development can optimize the system faster than trial and error iteration and reduce development time.

This chapter describes development efforts for an optical simulation system developed in MATLAB. It describes the results from the simulation routines applied to an optical system consisting of OAP mirrors, HW, and CPCs in both 2-D and 3-D.

5.2. 2-D simulation through HW

The first 2-D simulation to be performed was to vary the x position of the first and second OAP mirrors, which focus and collect the rays into and from the HW, respectively. The OAP mirrors used in the simulation have a focal length of 25.4mm and a diameter of 50.8mm. The resolution of the system was defined by the number of

vertices across the OAP mirror. The diameter of the OAP mirror was divided by 1025 points or pixels resulting in a pixel resolution of approximately 0.05mm/pixel. An obvious limitation of the pixel resolution is all the possible rays that can be estimated by the simulation can be estimate by launching 807 rays, one for each pixel in the 40mm diameter source.

Figure 5–1 diagrams 10 rays traveling through the HW setup when all rays are accepted and when they are rejected because of the HW's acceptance angle. The obvious affect of the acceptance angle is the limited portion of the beam that travels through the waveguide. In the 2-D simulation, 79% of the 40mm diameter source beam is rejected by the waveguide's acceptance angle criteria. The result is a reduction in throughput from approximately 94% to 20%. With the addition of the CPC (Figure 5–2), there is a slight improvement in throughput, 98% and 21% for simulations with and without the HW acceptance angle rejection, respectively.

The first experiment was to vary the x position of the focusing and collection mirrors. Both mirrors were moved from -5 to +5mm from the focal length position. Figure 5–3 shows the affects of x mirror positions on throughput and pathlength when all rays are accepted by the HW. As expected, the simulation shows the HW to be more robust to variation of the focusing mirror's x position because of the mathematical accuracy of the simulation. This created a well defined convergence to the focal point (Figure 5–4). Therefore, the mirror must be moved several millimeters before divergence causes the rays to miss the HW entrance. Because the HW is a non-imagine optical component

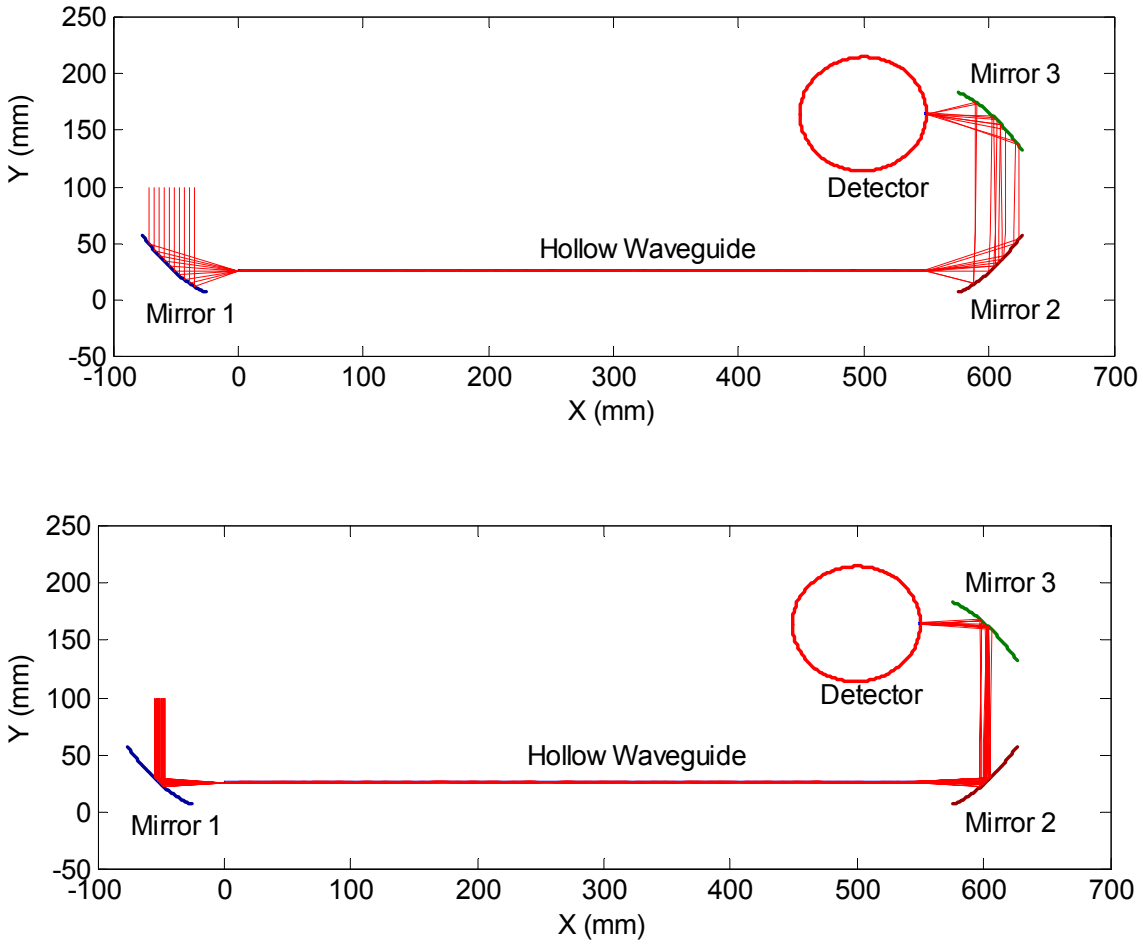

Figure 5–1. Example ray tracing of 2-D setup using three OAP mirrors. All rays accepted by HW (top) and with 15° acceptance angle criteria (bottom).

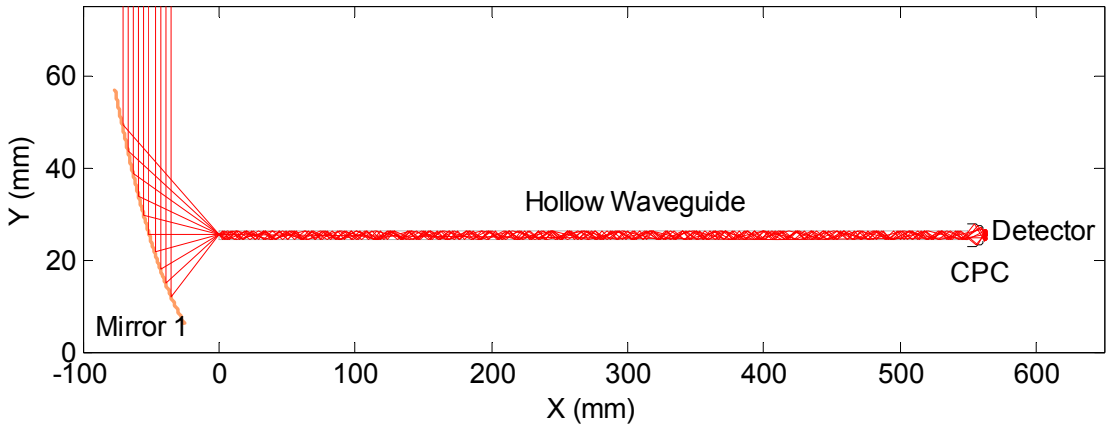

Figure 5–2. Example ray tracing of 2-D setup with CPC. All rays were accepted by HW.

(Figure 5–5), the collection mirror is more sensitive to the x position. The throughput starts to diminish within a fraction of a millimeter from the focal position. The maximum HW pathlength from the simulation is 562mm or 3% greater than the nominal length of the waveguide. When the acceptance angle is used for a ray rejection criteria (Figure 5–6), the maximum HW throughput decreases from 96% to 22% and the difference between the nominal length and the maximum HW length becomes minimal (<0.2% increase). Furthermore, inspection of the variation across mirror positions demonstrates the robustness on the system due to the acceptance angle restriction.

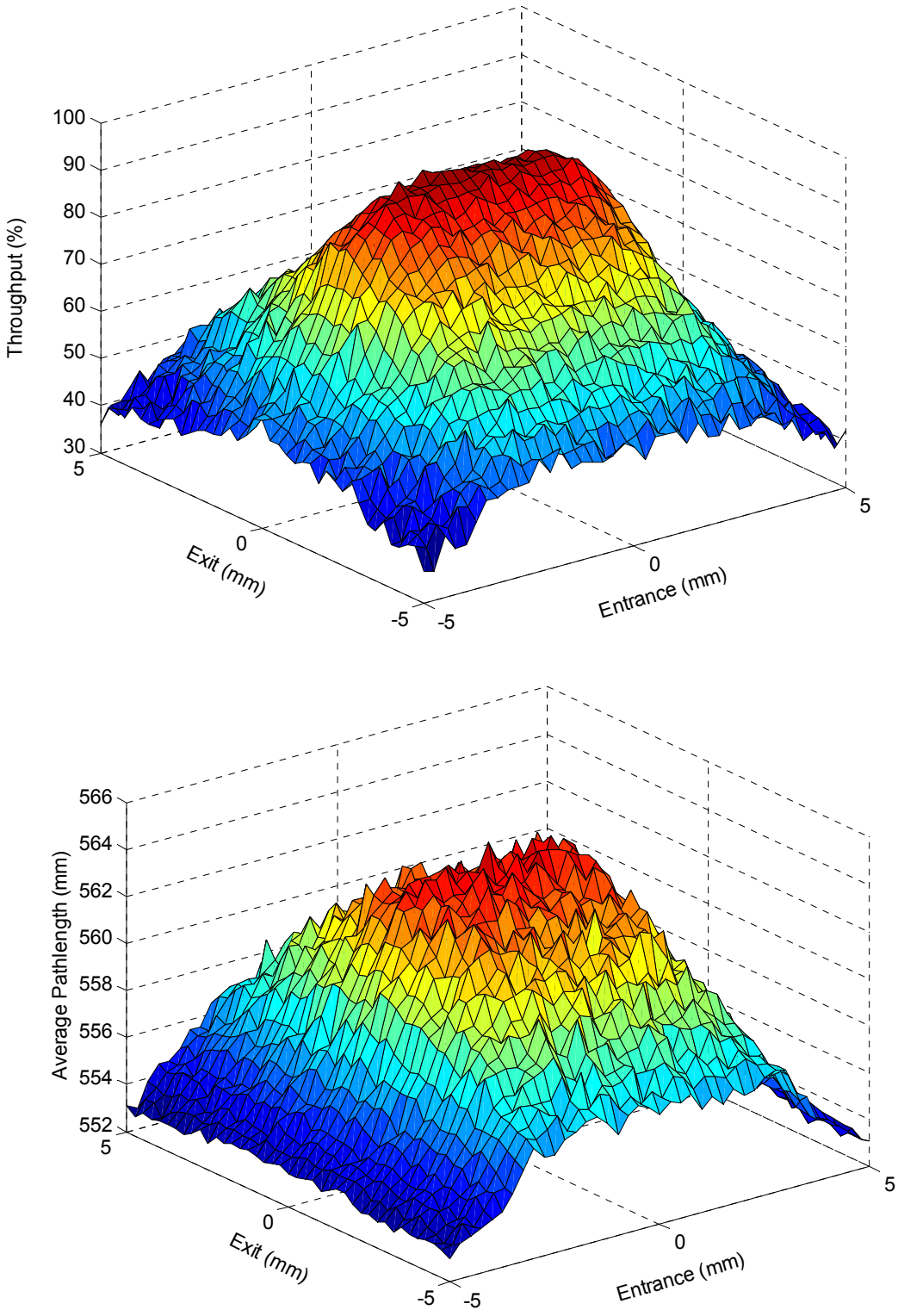

Figure 5–3. Two dimensional simulation of the effect of the x position of the entrance and exit mirrors on throughput (%) (top) and pathlength (mm) (bottom).

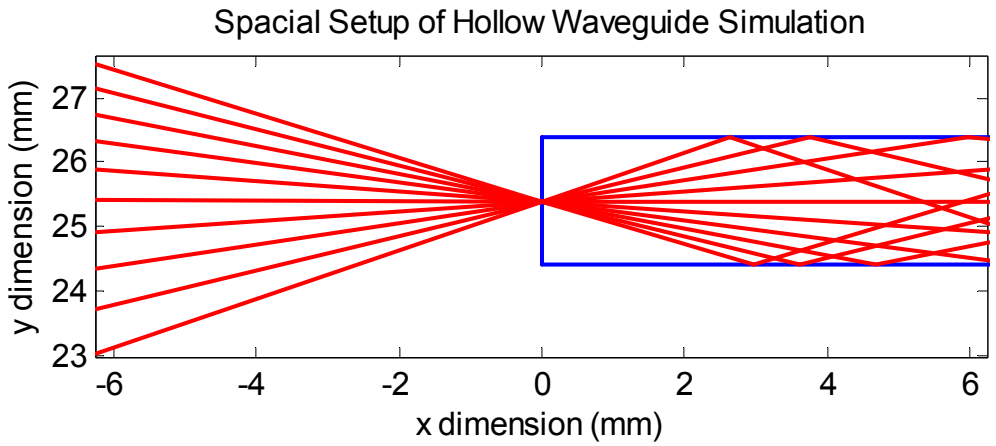

Figure 5–4. Two dimensional simulation of rays projected into the HW.

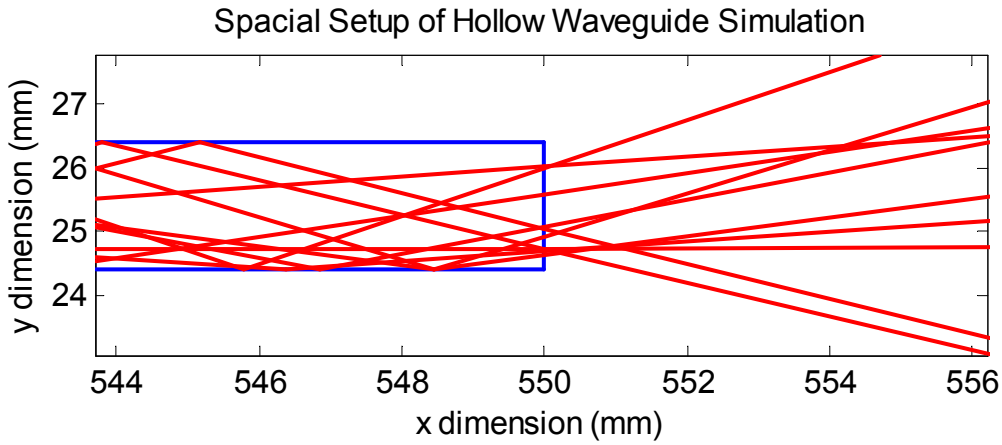

Figure 5–5. Two dimensional simulation of rays projected from the HW.

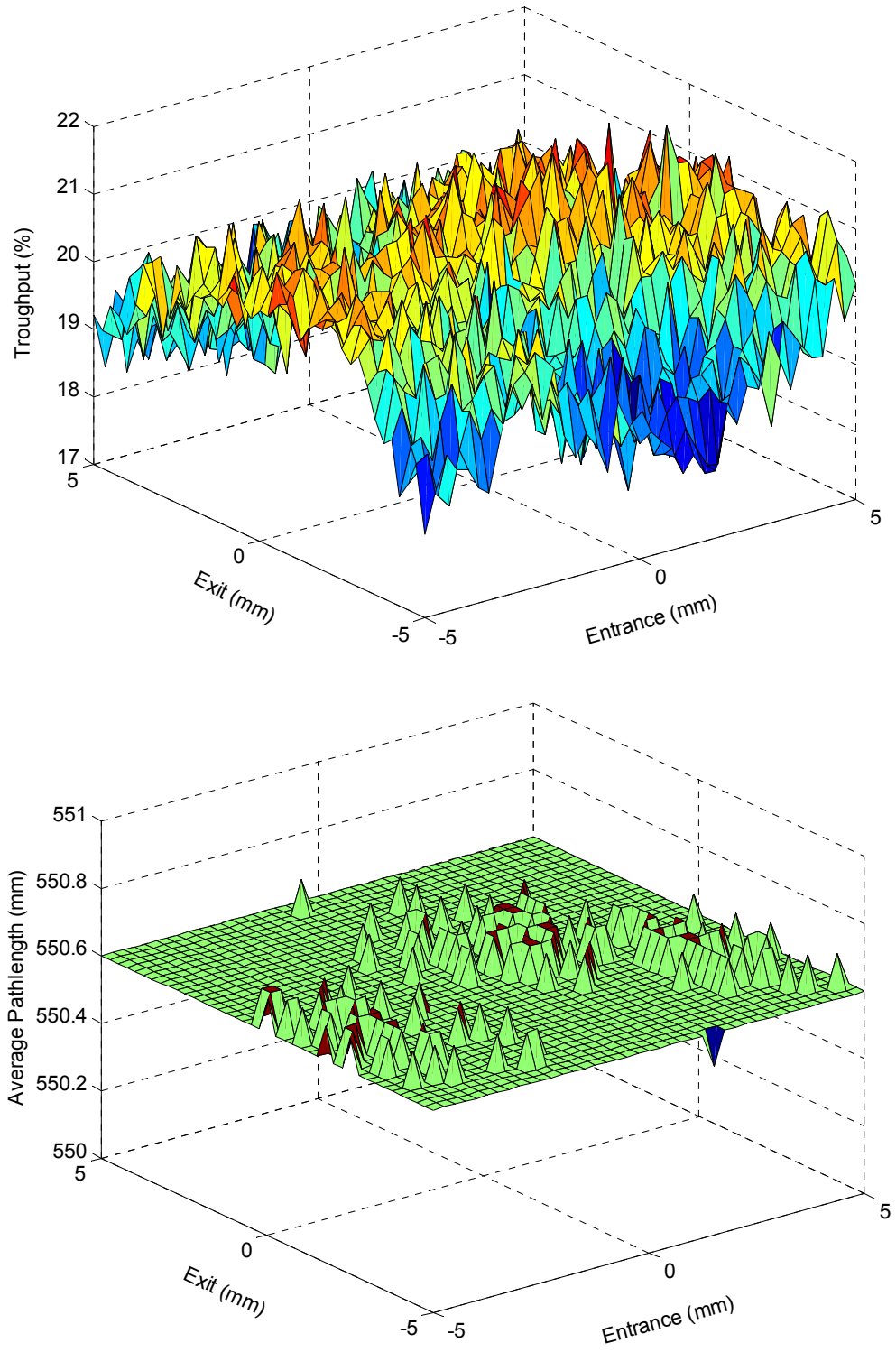

Figure 5–6. Two dimensional simulation of the effect of the x position of the entrance and exit mirrors and HW acceptance angle restriction on throughput (%) (top) and pathlength (mm) (bottom).

The second 2-D simulation experiment varies the xy position of the focusing OAP mirror. The experiments were performed to determine the influence of the skew rays on the throughput and pathlength of the HW. The simulation results for the experiment where all reflected rays were accepted are shown in Figure 5–7. The throughput contour plot shows the change in throughput becomes dramatic at approximately ±2mm from the focal distance along the x direction. The 2-D simulation indicates reasonable robustness to variation in the y direction ranging from approximately 88 to 97% throughput and 560 to 564mm pathlength. The reductions appear to be primarily due to rays not entering the waveguide (i.e. the rays miss the opening as the focal point approaches the edge of the opening) as opposed to the skew ray not being collected at the second OAP mirror. As expected, the pathlength contour plot shows maximum pathlength when the rays are focused at the center of the HW opening. Similar to the first set of experiments, applying the HW acceptance angle restriction reduces the throughput by nearly 80% and the pathlength is equivalent to the nominal length of the HW (Figure 5–8).

The application of the CPC adds new parameters to the simulation. In addition to the placement of the CPC, the entrance and exit apertures are variables affecting the design and efficiency of the concentrator. Two experiments were performed. First the CPC's distance from the exit of the HW was varied from 1 to 20mm as well as varying the size of the half size of entrance aperture from 1 to 5mm while the exit aperture was maintained at 0.8mm to maximize collection at the detector. Figure 5–9 and Figure 5–10 show the simulation results without and with the HW acceptance angle rejection, respectively.

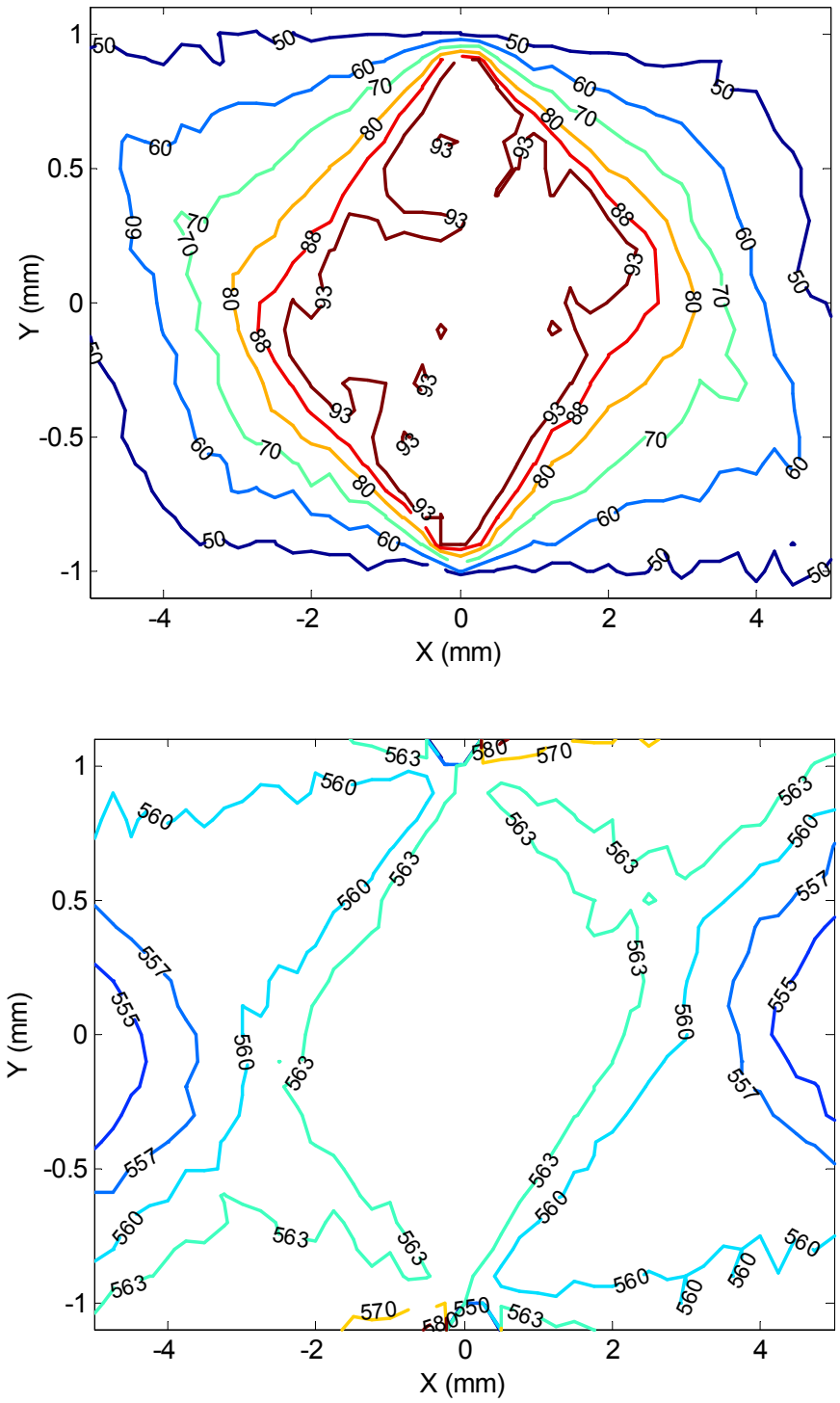

Figure 5–7. Two dimensional simulation of the effect of the x and y position of the entrance mirror on throughput (%) (top) and pathlength (mm) (bottom).

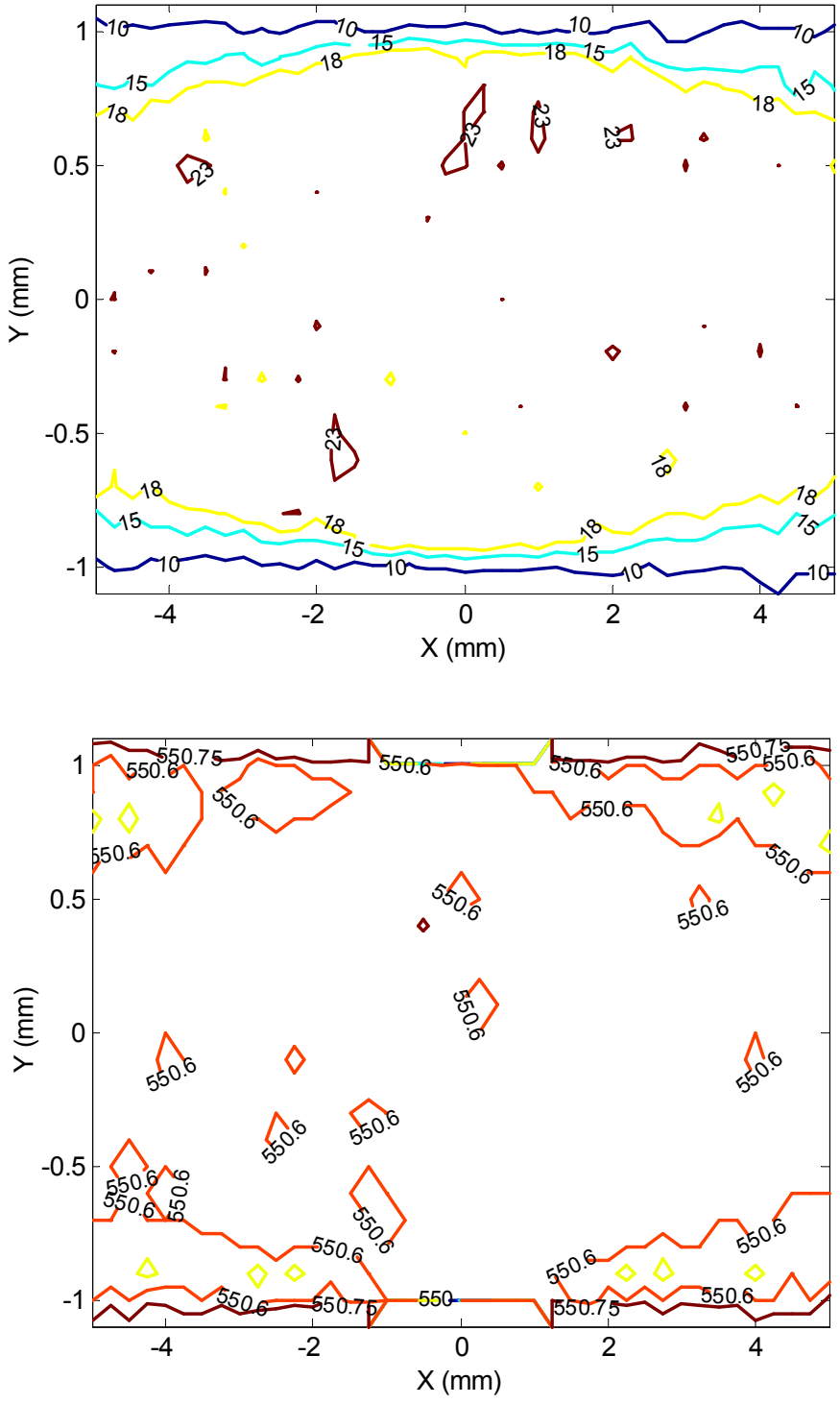

Figure 5–8. Two dimensional simulation of the effect of the x and y position of the entrance mirror and HW acceptance angle on throughput (%) (top) and pathlength (mm) (bottom).

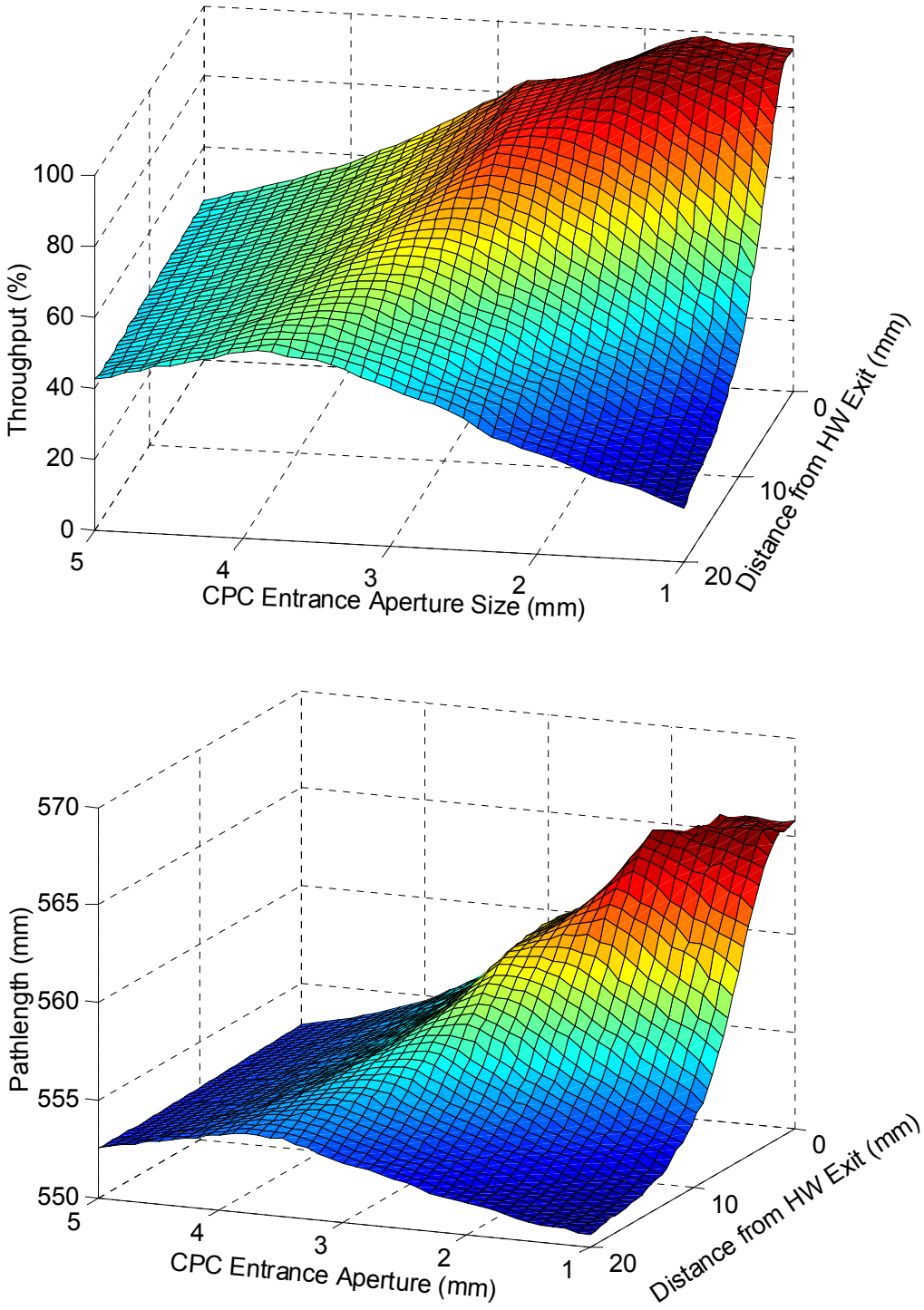

Figure 5–9. Two dimensional simulation of the effect of the CPC entrance aperture and distance from HW on throughput (%) (top) and pathlength (mm) (bottom).

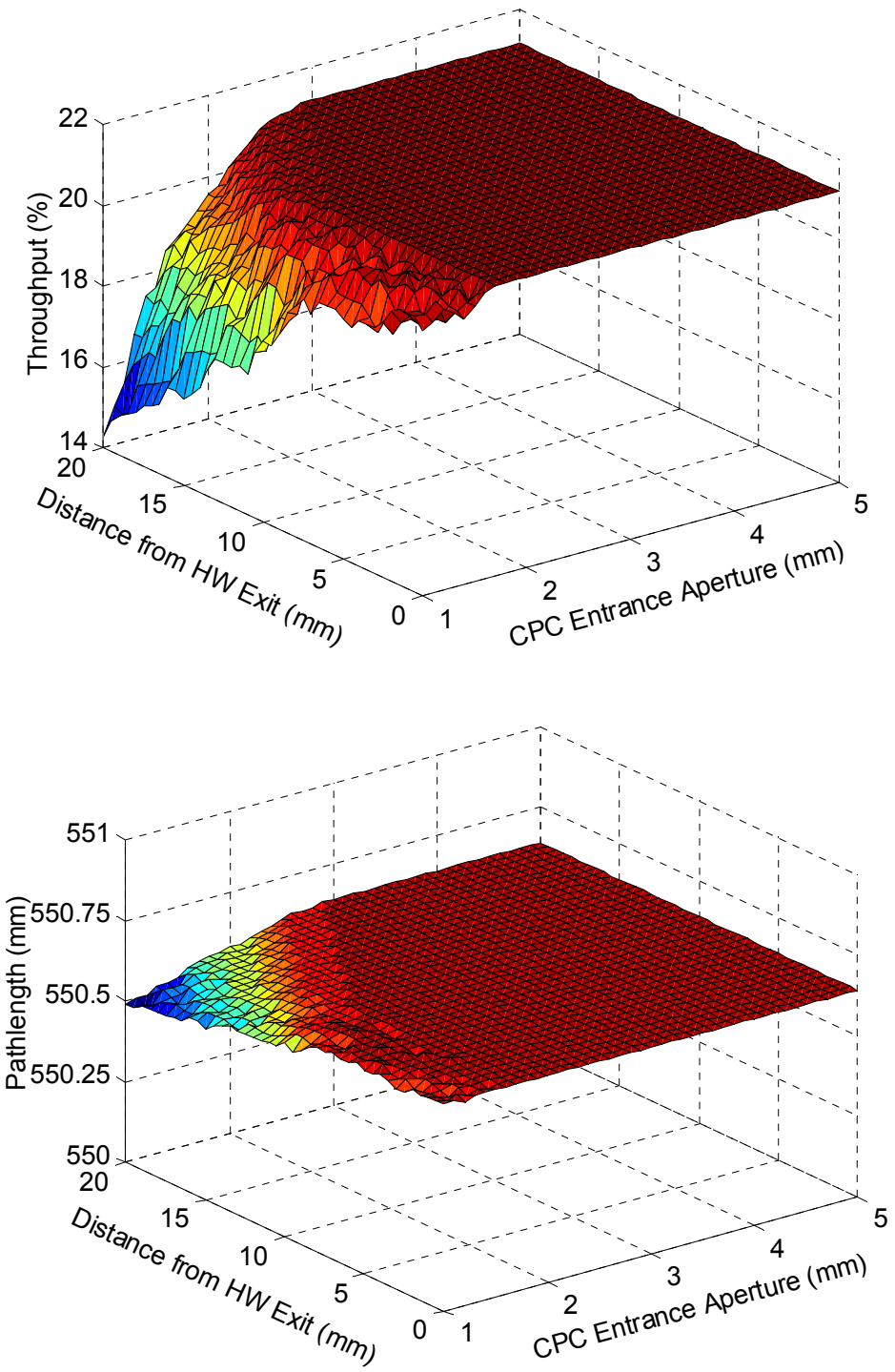

Figure 5–10.Two dimensional simulation of the effect of the CPC entrance aperture and distance from HW and HW acceptance angle on throughput (%) (top) and pathlength (mm) (bottom).

Given the various angles the rays travel from the HW when the acceptance angle criteria is not used, the reduction in throughput as the CPC is moved further away from the HW is expected. The reduction in throughput with increased entrance aperture indicates the efficiency of the CPC is related to the ratio of the entrance to exit aperture. The larger the ratio of the entrance to exit aperture, the steeper the curve is on the CPC interior surface and the greater the potential is for the ray to be reflected out of the CPC. Consummate with the reduction in throughput is the reduction in pathlength. As the CPC is moved further away from the exit of the HW, fewer of the rays with large angles and longer paths through the HW are within the entrance diameter of the CPC. This is analogous to diminished lighting when walking further into a cave. As seen in the previous experiments, the application of the acceptance angle criteria both reduces throughput and pathlength and improves robustness to optical component positioning. Comparing the performances of the CPC and the collection mirror when varying their position, the simulations indicate utilization of the CPC can result in a system that is less sensitive to variation in position the optical components, as well as a more compact design.

The final 2-D experiment positioned the CPC at 10mm from the exit of the HW and varied the entrance and exit aperture half sizes. This was to simulate the current physical arrangement (*i.e.* because of the HW mounting, the closest position is approximately 10mm from the HW exit) and investigate optimizing the CPC design. The focusing OAP mirror was set at the optimum position, x distance equal to the focal length and y set to the center of the HW. Figure 5–11 shows the results for the simulation when the HW

accepts all the incoming rays. The maximum throughput is approximately 80%. It occurs when the half sizes of the entrance and exit apertures of the CPC are 3 and 0.45mm, respectively. The pathlength at this point is 560mm, a 2% increase over the nominal length of the HW. The figure shows there is an optimum set of apertures for the application. It also demonstrates the decrease in performance as the ratio of the apertures increases. Figure 5–12 shows the simulation results with the HW acceptance angle rejection applied. The throughput plot shows an optimum area of nearly maximum throughput for the attenuated beam (approximately 21%). The ranges for the apertures are approximately 2 to 8mm for the entrance and 0.1 to 0.5mm for the exit. The pathlength for these simulations were not appreciably different than the length of the HW.

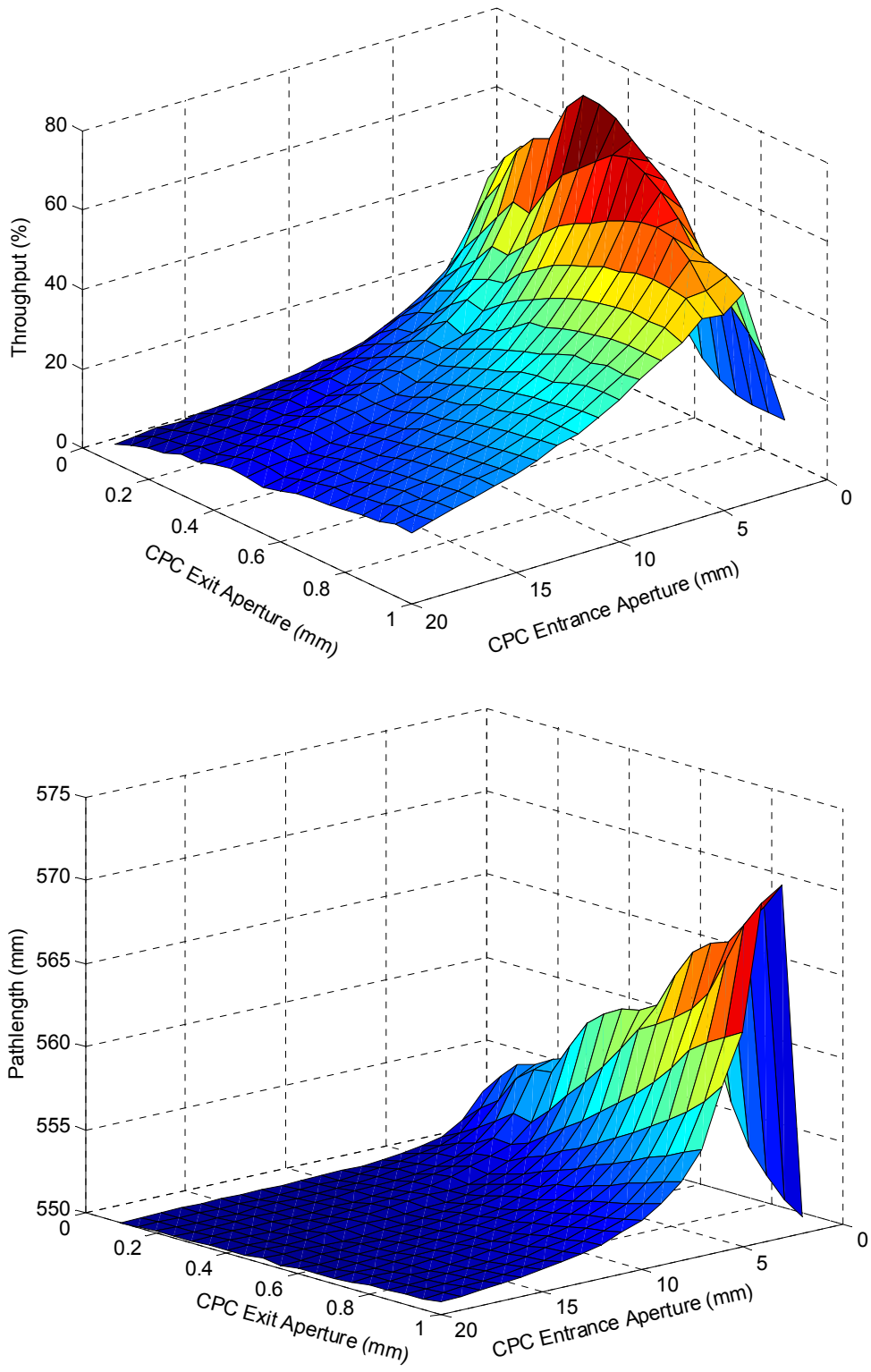

Figure 5–11.Two dimensional simulation of the effect of CPC apertures on throughput (%) (top) and pathlength (mm) (bottom) of HW setup.

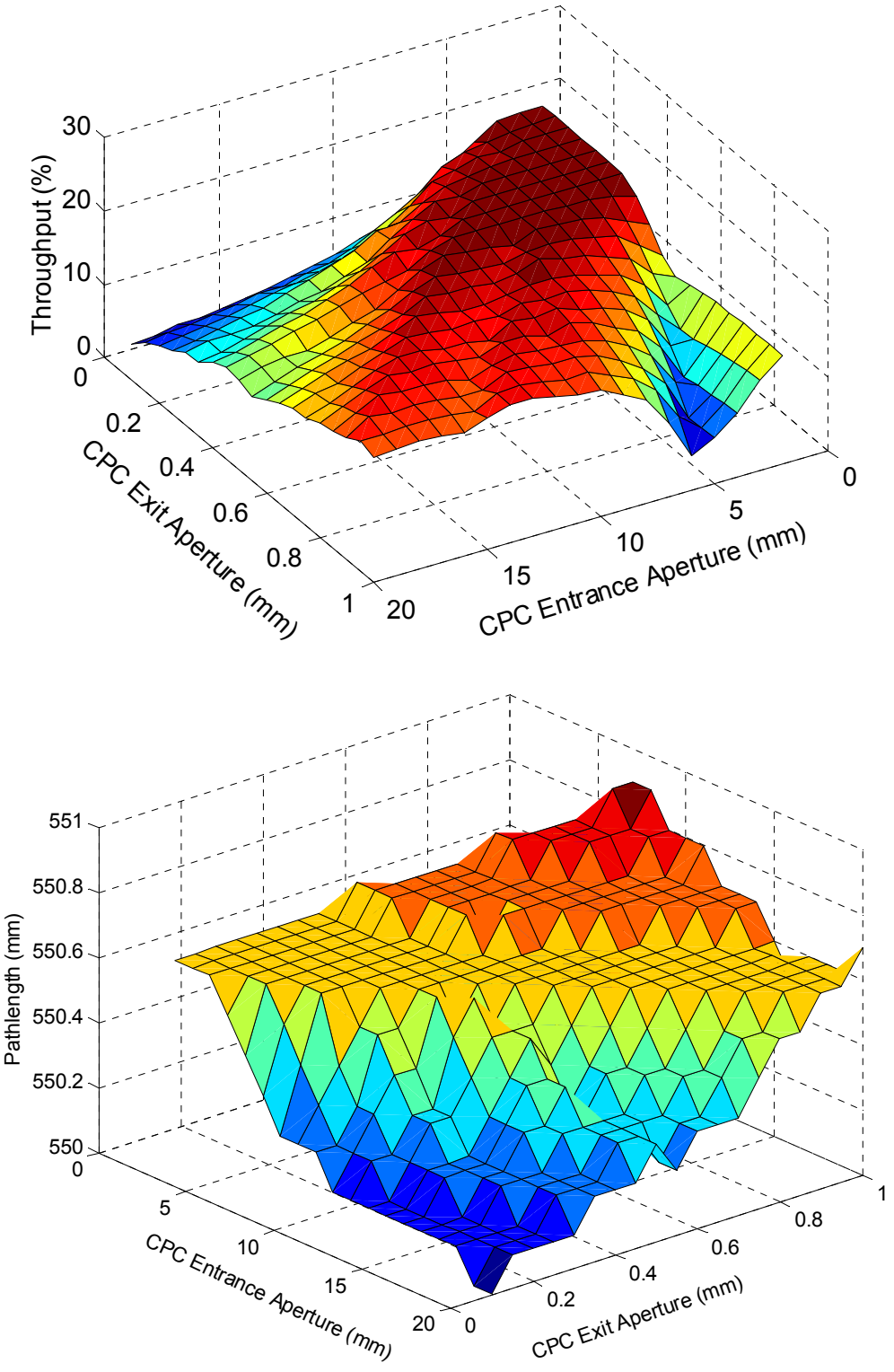

Figure 5–12.Two dimensional simulation of the effect of CPC apertures and HW acceptance angle on throughput (%) (top) and pathlength (mm) (bottom).

5.3. 3-D simulation through HW

The addition of the third dimension demonstrated some profound affects. First is the increase in computation time. In the 2-D simulations with 807 rays the computation time ranged from 0.1 to 0.4 minute on computers with 1.7GHz processors. Similar simulations with 1000 rays in 3-D on the same computers required 0.5 to 15 minutes. Another aspect of ray propagation through waveguides evident from simulations is the corkscrew progression of the skew rays (Figure 5–13). The meridional travels a well behaved path through the waveguide because the plane of incidence at all the reflection points includes the axis of rotation of the waveguide. In the case of the skew rays, the plane of incidence forms an angle to the axis or rotation resulting in reflections that travel side to side as much as they propagate through the waveguide. Finally, the number of potential launching points from the source beam increases from 807 in the 2-D simulation to over 5×10^5 in the 3-D simulation. Therefore, the 1000 rays launched is a relatively small percentage of the potential rays, but demonstrates the power and utility of the Monte Carlo simulation.

The addition of the CPC also affected computation time. In cases when the ratio of entrance to exit aperture was large, permitting steep angles of incidence and nearly flat surface at the entrance of the CPC, the number of bounces within the CPC could increase by an order of magnitude. The result is computational times up to 60 minutes for 1000 rays. Further research into simulating a CPC may yield an improved method of predicting ray paths to minimize the computation time. Figure 5–14 shows a sampling of

rays traveling through a CPC with entrance and exit apertures of 2 and 0.4mm, respectively.

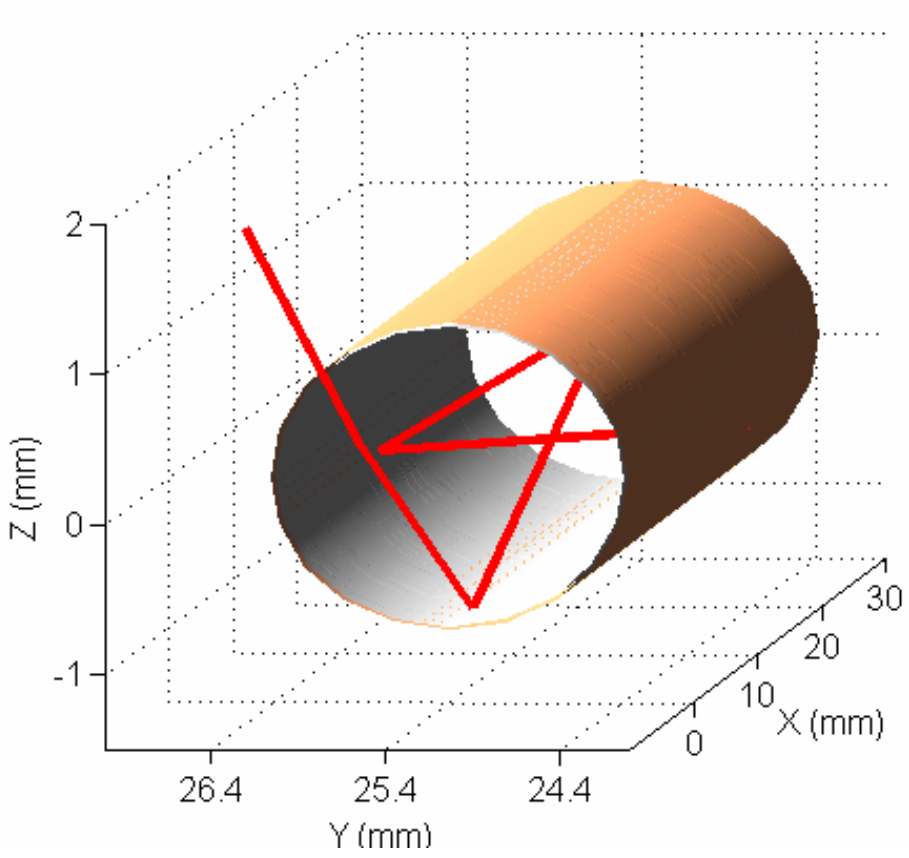

Figure 5–13. Diagram of simulated ray traveling in a corkscrew fashion through the HW.

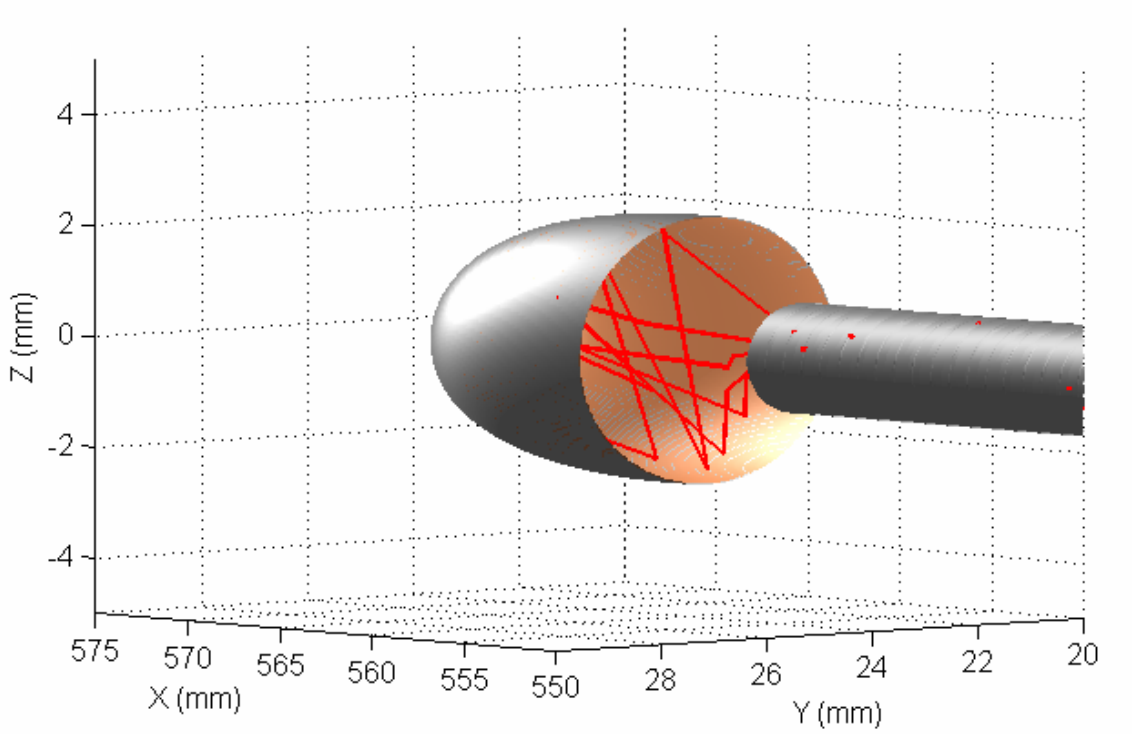

Figure 5–14.Example of five rays propagating through a CPC with entrance and exit apertures of 2 and 0.4mm, respectively. (Detector not shown.)

The first 3-D simulation experiment measured the effect of y and z positions of the focal point from the first OAP mirror on HW throughput and pathlength. The y and z positions were varied from one edge of the HW to the other (i.e. -1.1 to 1.1mm). The results without and with consideration of the acceptance angle of the HW are shown in Figure 5–15 and Figure 5–16. The estimates for throughput when all rays are propagated are similar to the 2-D simulations, approximately 93%. When the acceptance angle is considered, the throughput in the 3-D simulation reduces by nearly an order of magnitude of the 2-D simulation to approximately 3%. A 3-D simulation testing all the originating rays shows only 3% of the rays have angles less than the 15° acceptance angle for the

HW. The pathlength estimates from the 3-D simulation are about the same as the 2-D simulation with and without ray rejection due to HW acceptance angle, approximately 580mm and 551mm, respectively. These results, in particular the throughput measurement, demonstrate the need to include the third dimension at the expense of computation time in optical simulations.

The second set of experiments utilized the CPC. As with the 2-D simulation, the first experiment with the CPC varied the distance from the exit of the HW and the CPC entrance aperture. The distance was varied from 1 to 20mm, as was done previously, but due to computation time differences; the aperture half size was varied from 2 to 7.8mm. The focusing mirror was optimized to the focal length of the mirror at the center of the HW. The exit aperture of the CPC was fixed at 0.8mm. The results for simulations where none of the rays were rejected are shown in Figure 5–17. The throughput plot shows greater sensitivity to the distance from the exit of the HW; whereas, the pathlength appears to be consistent with the 3-D yz position simulations.

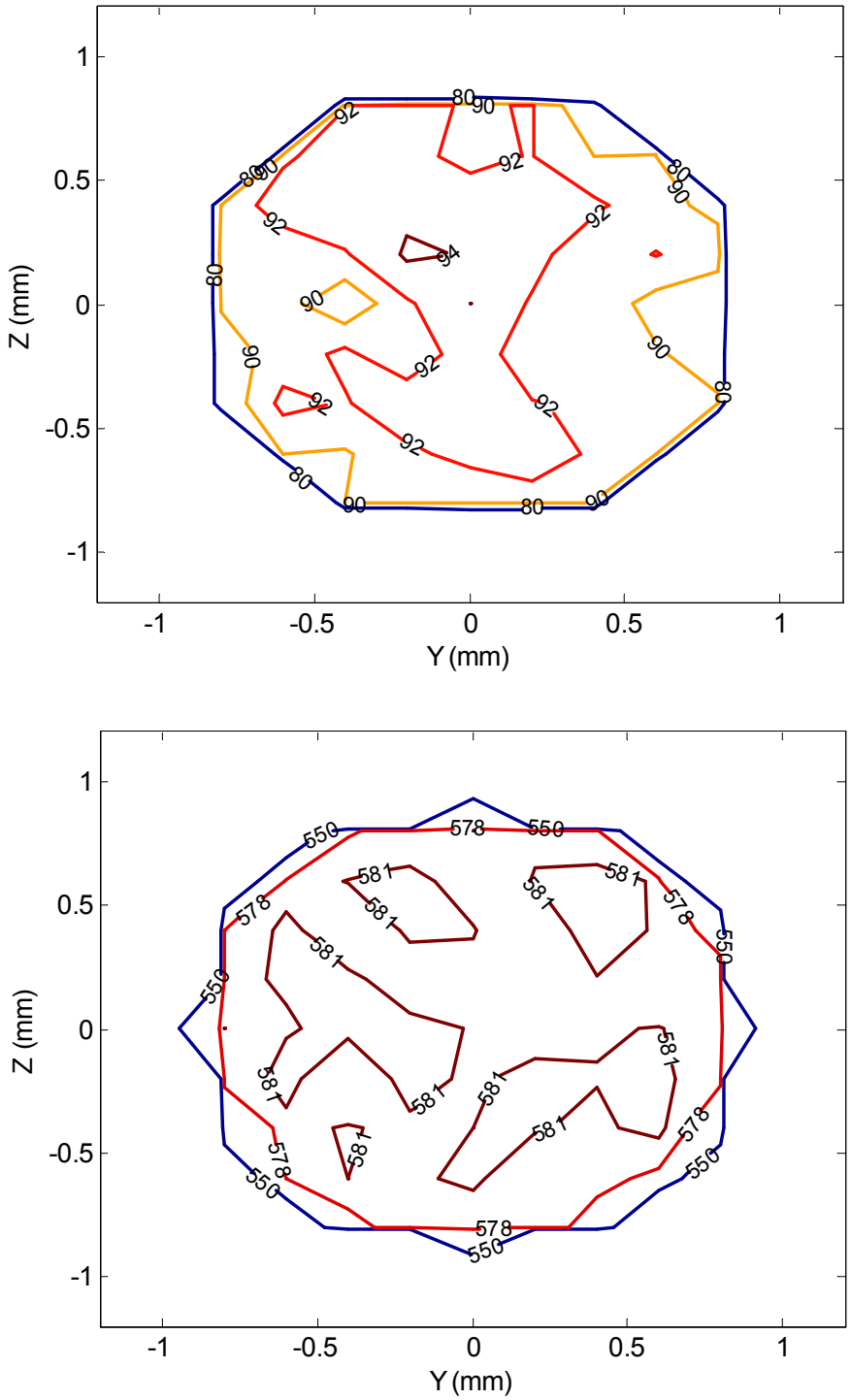

Figure 5–15.Three dimensional simulation of the effect of y and z position of OAP mirror 1 on throughput (%) (top) and pathlength (mm) (bottom) of HW setp.

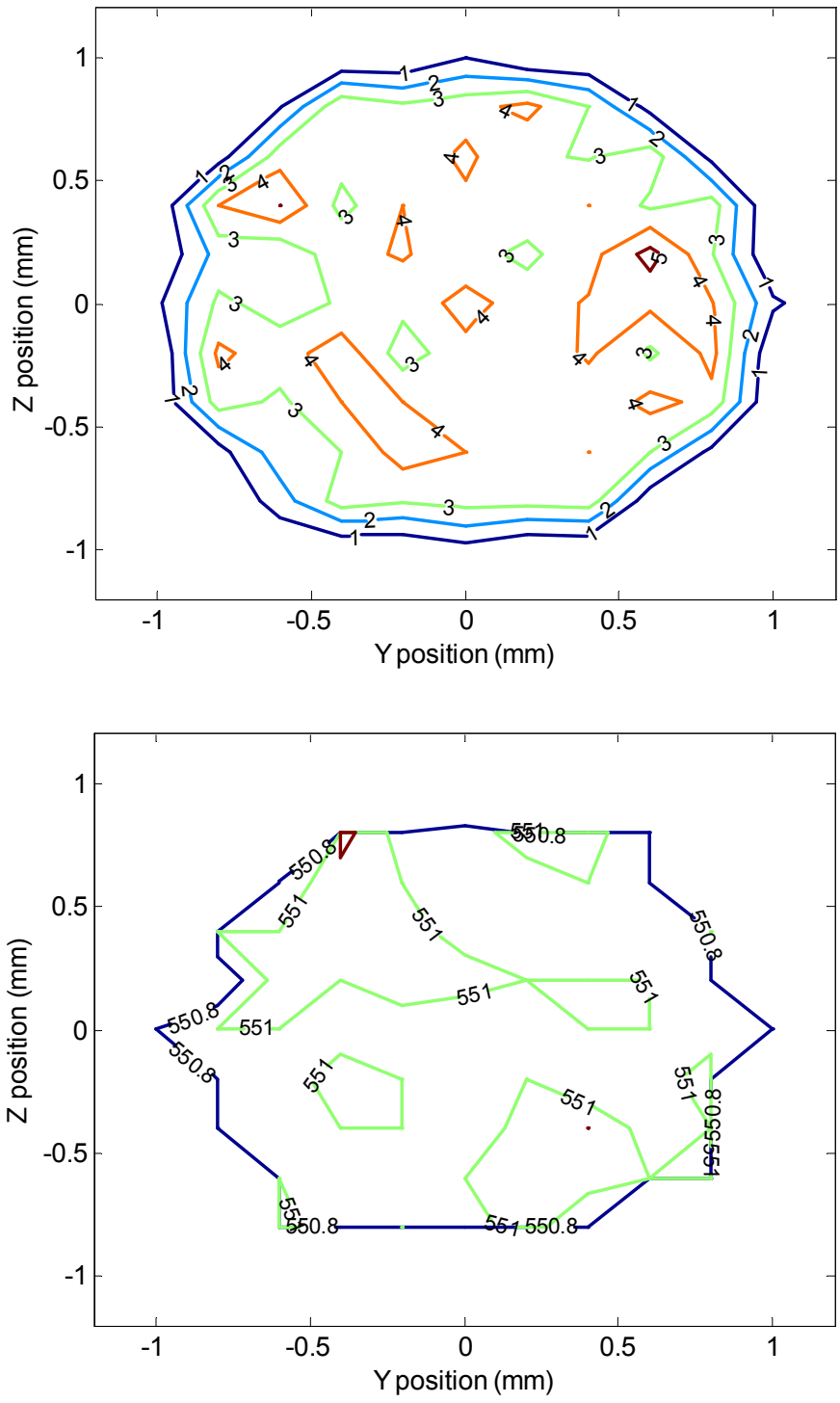

Figure 5–16. Three dimensional simulation of the effect of y and z position of OAP mirror 1 and HW acceptance angle on throughput (%) (top) and pathlength (mm) (bottom) of HW setup.

The results for simulations including the HW acceptance angle criteria shows consistent performance across the variables and are approximately the same as the results using OAP mirrors for collection and focusing (Figure 5–18). Even though the figures show variation across the variables, the scale is quite narrow. The ranges were <4% for throughput and <0.5mm for pathlength. The results indicate that there would not be optical performance deterioration by utilizing the CPC. The comparable performance is probably due to the rays exiting the HW at moderate angles and not on steeply divergent paths. Therefore, adapting the CPC to the detector as a separate optical component from the HW is feasible.

The third 3-D experiment varies the entrance and exit apertures of the HW. The half distance of the entrance aperture (*a*) was varied from 2 to 7.8mm, while the half distance of the exit aperture (*a'*) was varied from 0.1 to 1mm. Figure 5–19 shows the affect of these parameters when all rays are accepted by the HW. Like the previous experiments, both the throughput and pathlength are consistent across the variables and comparable to the previous experiment. Higher throughput is achieved with a larger entrance diameter on the CPC to collect the steeply divergent rays. This is evident from the fairly uniform result across the range of exit apertures in the throughput plot in Figure 5–19. Figure 5–20 shows a different effect with respect to CPC entrance and exit apertures when the HW acceptance angle is taken into consideration. The magnitude of the throughput and pathlength are comparable to the previous experiment. The throughput plot shows consistency of results for the range of entrance aperture and slight sensitivity to the exit aperture size. The lack of divergent rays is probably responsible for the observed effect.

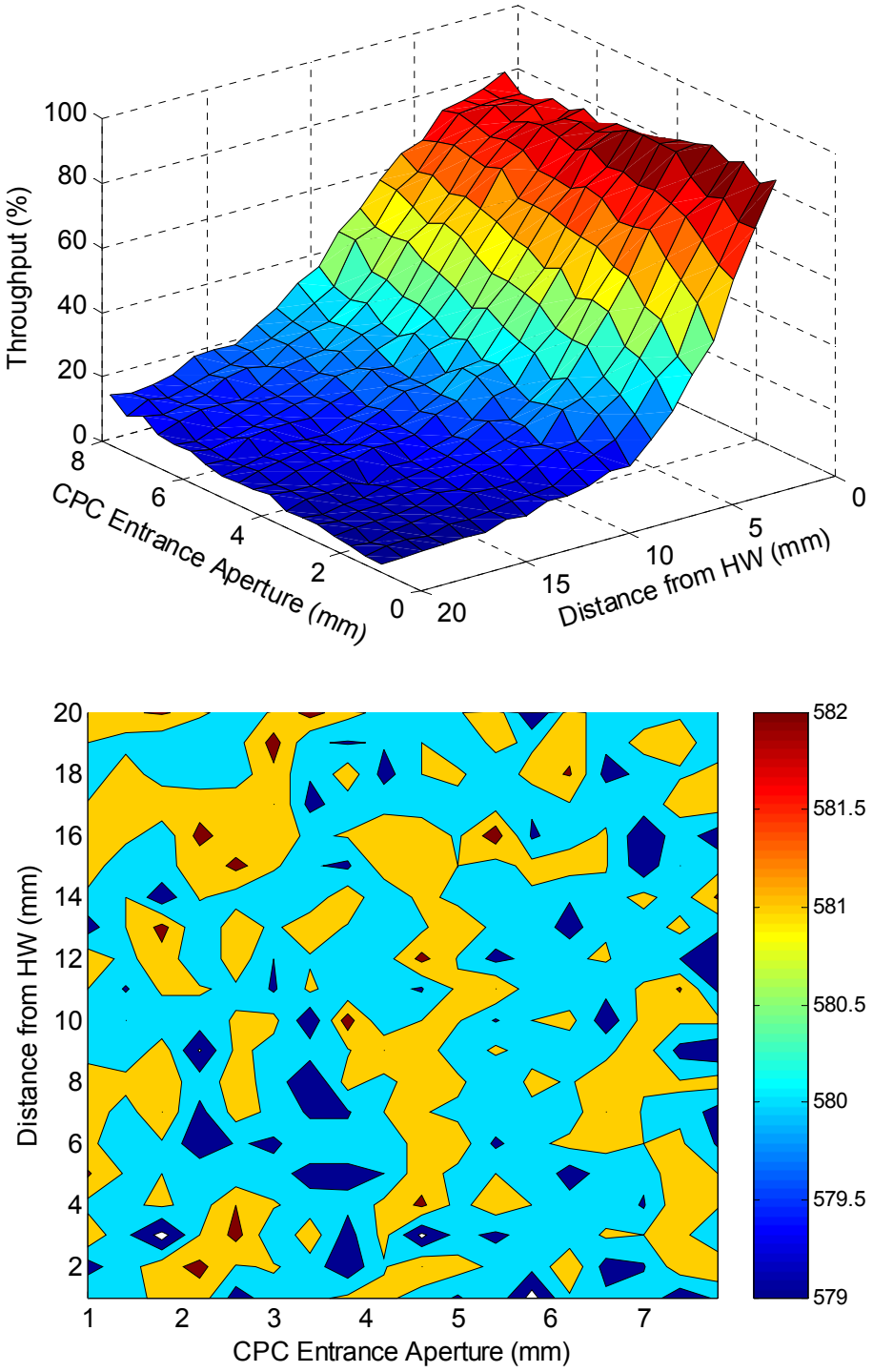

Figure 5–17. Three dimension simulation of the effect of CPC entrance aperture and distance from HW exit on throughput (%) (top) and pathlength (mm) (bottom).

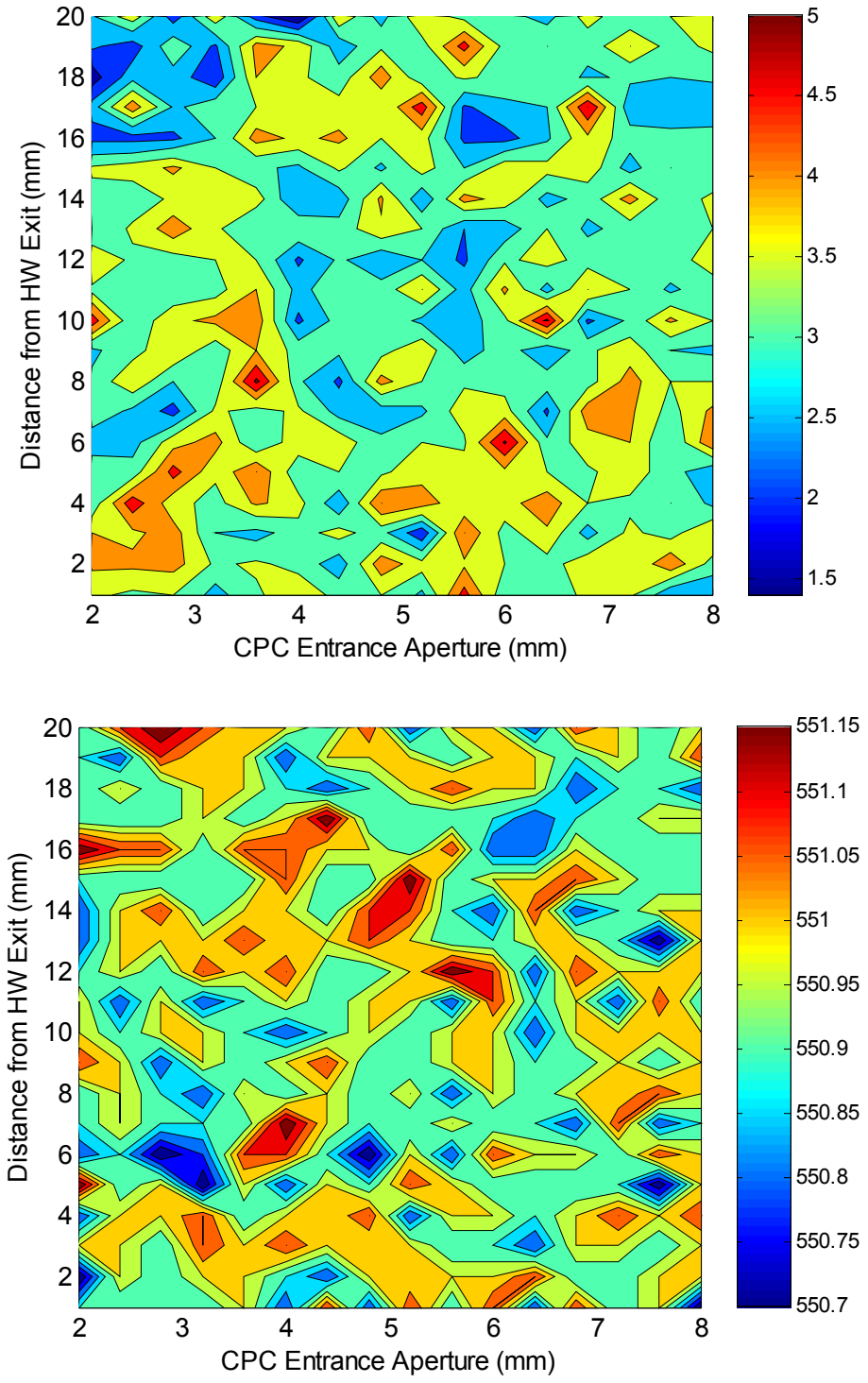

Figure 5–18. Three dimension simulation of the effect of CPC entrance aperture and distance from HW exit and HW acceptance angle on throughput (%) (top) and pathlength (mm) (bottom).

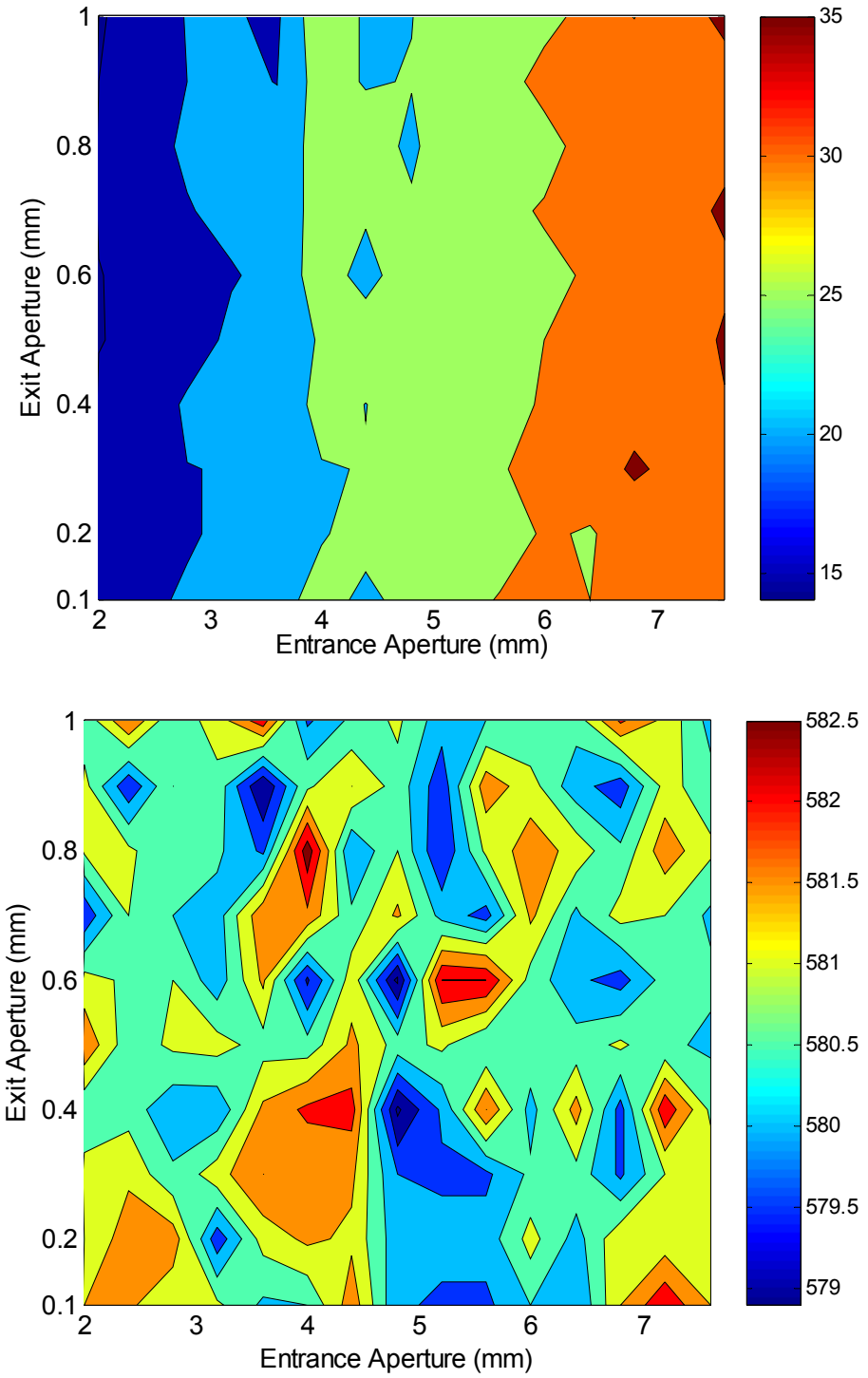

Figure 5–19.Three dimension simulation of the effect of CPC entrance and exit apertures on throughput (%) (top) and pathlength (mm) (bottom).

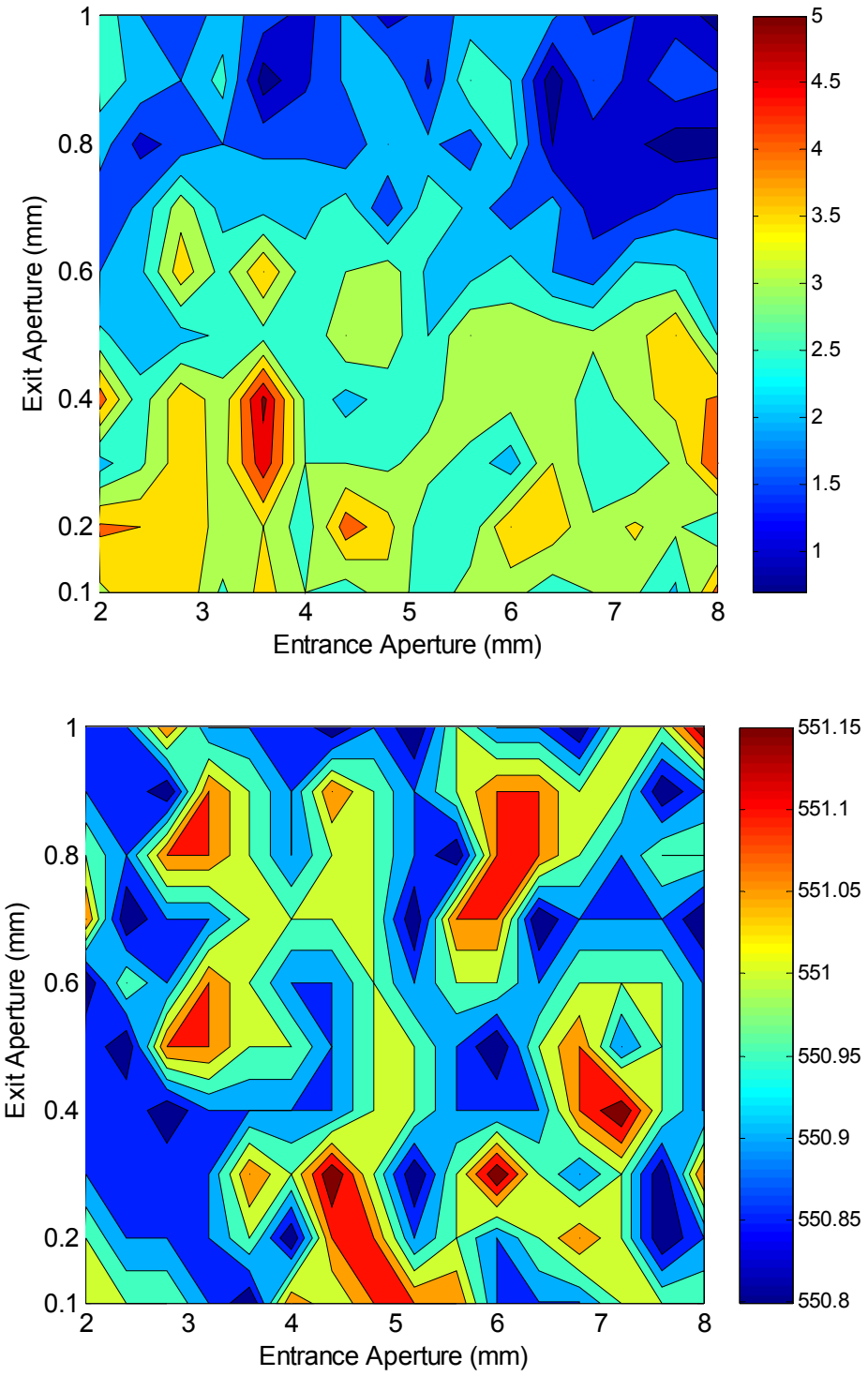

Figure 5–20. Three dimension simulation of the effect of CPC entrance and exit apertures and HW acceptance angle on throughput (%) (top) and pathlength (mm) (bottom).

5.4. Evaluation of Optical Configuration

Developing improvements to the optical configuration of the sensing system is a goal of modeling and ray tracing. In Section 3.3, the sensitivity of the HW was estimated to be a 125 fold increase over the multi-pass gas cell because of optical efficiencies, but only 60% was realized. Simulation experiments were performed in 3-D to estimate the magnitude of energy loss to the acceptance angle of the HW. The result of the experiment was only 3% of the rays launched were propagated into the HW. Using this estimate, the 125 fold increase in sensitivity was reduced to 1.2, a decrease of an order of magnitude.

Improvements in the sensing system could be realized when more of the IR is accepted into the waveguide. The acceptance angle of the waveguide is a function it the construction materials and geometry. There would be additional trade-offs (*e.g.* attenuation) by using other materials. A better approach would be to improve the condensing optics to produce a beam with a less steep convergence of rays. This can be simulated by using OAP mirrors with longer focal lengths. A series of 3-D simulations were performed with focusing OAP mirrors with focal lengths varying from 25.4 to 127mm. Each simulation launched a million rays and determined angle along the x-axis of the reflected ray. If the angle was less than or equal to 15°, it was counted as accepted by the HW. The results are listed in Table 5–1.

Table 5–1. Throughput results for 3-D simulations of various focal length. OAP mirrors

Focal Length	Throughput (%)
25.4	3.6
50.8	14.2
76.2	31.8
101.6	56.5
127.0	85.3

The results from the simulation show an improvement of over an order of magnitude by using a longer focal length mirror. The use of an OAP mirror with a focal length of 127mm is impractical for sensor development because it would increase the length of the sensing system by 50%. Other optical configurations can be designed to achieve the function of the longer focal length mirror in a more compact system. Given the potential 120% increase in throughput, the drawback of energy losses due to reflections should be of minimal concern. The simulation routines can be used to evaluate various multiple mirror configurations.

Some potential optical configurations include the inverse of a beam expander or collection optics similar to what is used in telescopes. Considering the 15° acceptance angle, the numerical aperture of the waveguide is approximately 0.25. Theoretically, a focusing configuration with an f/# of 2 should maximize the energy collection into the waveguide.

5.5. Conclusion and Outlook

Optical simulations can provide valuable information in designing compact optical sensors. Non-commercially available optical components, like the HW and CPC were

successfully modeled to estimate throughput and pathlength while varying the position or size of other optical components. Both 2-D and 3-D simulations were performed. The 3-D simulation was more computationally expensive by one to two orders of magnitude. The 3-D simulation also showed the throughput to reduce by an order of magnitude over the 2-D simulation with consideration for HW acceptance angle. Furthermore, the simulations indicate that sensitivity of the system could be improved by an order of magnitude of more by improving the coupling of the IR beam into the waveguide. The application of the CPC to collect the rays from the HW and focus on the detector does not require intimate (<10mm) proximity to the HW exit. Therefore, the CPC can be practically applied in this setup with entrance and exit apertures of approximately 3.5 and 0.4mm, respectively.

Simulations can be an important tool in optical sensor development. Future directions for simulation technology are to optimize ray tracing for the CPC and including a spectral prediction component. The first objective could be accomplished by determining something comparable to an acceptance angle for the CPC. The latter could be accomplished with SPRAY and SCOUT commercial software from M. Theiss Hard and Software. The SPRAY software has ray tracing capability for the optical components. It has the capability of assigning optical materials to the surfaces to estimate the spectral result for a given optical configuration. The SCOUT software can be used to generate the spectral characteristics of optical materials. For example, experimental data for CO and NO absorption bands can be fitted with a series of curves that model a harmonic oscillator. The simulated spectral response can be used as the materials inside the HW

and spectral responses can be predicted for various configurations. The configurations would not be limited to the HW. The multi-pass gas cell can be modeled and compared with the HW and the experimental results.

APPENDIX A.

MATLAB SCRIPTS AND FUNCTIONS

Name	Script or Function	Desciption
collectfromanalyzers	script	Collect analog data from analyzers and flow meters
ftirconv3	script	Computes mg of CO or NO when given ppm, time increment and flow rate.
loadext	script	Extracts loadings saved from a quant2 calibration
opusdde02	function	Extracts data blocks from OPUS
isok	function	Checks for "OK" as first line of a string
analconv04	function	Converts analyzer analog data to concentration
compstat4	function	Computes t-test and RMSEP results for data comparison
taucalc	function	Calculates the time constant for a decay curve
findcoords3	function	Finds the x,z coordinates of the closest array element
hw55cmwca	script	Simulates 2-D ray trace through hollow waveguide with a compound parabolic concentrator at the detector
HWdet	script	3-D coordinates of the detector element
HWdetwc	script	3-D coordinates of the detector element with the compound parabolic concentrator
hwgg55cmg	script	2-D simulation ray trace of 55cm hollow waveguide with off axis parabolic mirrors
HWhw02	script	Generates the 3-D hollow waveguide
HWoapm1d	script	Generates the 3-D off-axis parabolic mirror in position one
HWoapm2d	script	Generates the 3-D off-axis parabolic mirror in position two
HWoapm2d	script	Generates the 3-D off-axis parabolic mirror in position three
HWsource01	script	Defines the 3-D source position
testHWsim14	script	3-D ray trace simulation of hollow waveguide with off-axis parabolic mirrors
testHWsim15wc	script	3-D ray trace simulation of hollow waveguide with compound parabolic concentrator

APPENDIX A-1. COLLECTFROMANALYZERS.M

```
% Collectfromanalyzers script collects analyzer data through the Newport ESP6000
% motor controller board, which has 8 analog inputs.  The user must
% manually set the variable name to save data to.  The program initializes
% the stage, then waits for the user to begin data collection by pressing a
% key (typically the spacebar).  The program continues to collect data
% until the user presses ctrl-c to abort the program.  The user then
% manually saves the data (highlight line, right click, evaluate
% selection).  Inputs 0 through 5 are set to NO, CO, CO2, total flow, HW
% flow.
%
% Written by Bruce Thompson, 4/22/03

m3_cal07std15 = zeros(2500,7);
initstage;
disp('Press any key to start');
pause;

sout = sprintf('   Time      Cycles      NO       CO       CO2       F1       F2');
disp(sout);

i = 0;
t_begin = clock;
t_stop = t_begin;
while i<=2500
   i = i + 1;
   t_start = t_stop;
   while etime(t_stop,t_start)<1.0
      t_stop = clock;
   end;
   [V,T] = collectadcdata8;
   m3_cal07std15(i,:) = [etime(t_start,t_begin),T,V(1:5)];
   sout = sprintf('%7.2f        %10d        %7.4f        %7.4f        %7.4f        %7.4f
%7.4f',etime(t_start,t_begin),T,V(1:5));
   disp(sout);
end;
m3_cal07std15 = m3_cal07std15(1:i-1,:);
save m3_cal07std15 m3_cal07std15
```

APPENDIX A-2. FTIRCONV3MF.M

```
% outarray = ftirconv3mf(inarray,cmpd,deltat,F);
% computes mg CO when cmpd=1
% computes mg NO when cmpd=2
% deltat = time increment, either one for each row entire inarray.
% F is the flow rate, agian one per row or one per array.

% Written by B. T. Thompson, 12/1/03

function outarray = ftirconv3mf(inarray,cmpd,deltat,F);

%convert flow, F, from L/min to L/sec
F = F/60;
% molar volume at 25C and 1atm
%Vm = 24.465;
% molar volume at 70F (23.9C) and 1atm
Vm = 24.3729525;
[r1,c1] = size(inarray);
[r2,c2] = size(cmpd);
if r2>1
   cmpd = cmpd(1,:);
end;
if c1~=c2 & c2~=1
   error('Input array and compound declaration array are not of the same number of columns!');
end;
inarray(find(inarray<0))=0;
outarray = inarray;

if c2==1
         if cmpd==1
      MW = 28.01;
         else
      MW = 30.01;
         end;

         outarray = inarray.*deltat.*F*MW/(1000*Vm);
else
   for i = 1:c2
         if cmpd(i)==1
         MW = 28.01;
```

```
            else
        MW = 30.01;
            end;

            outarray(:,i) = inarray(:,i).*deltat.*F*MW/(1000*Vm);
   end;
end;
```

APPENDIX A-3. LOADEXT.M

```
% extract loadings from OPUS
%script extracts the pls loadings that were saved from a quant2 calibration
%the file path and name for the loadings must be manually updated in the file
%Written by Bruce T. Thompson, 11/19/2003



filepath = 'C:\PhD Research\3m cell\2cmresol\Calib0007r2\Temp\';
filename = '3m_CO Q2r2all01_3m_CO Q2r2all01';
PLSext = '_PLS_V';
loading = '1';
dot0 = '.0';
loadname = 'm3_COQ2r2all';

for lnum = 1:10
        loading = num2str(lnum,'%1d');
        name = [filepath,filename,PLSext,loading,dot0];
        eval([loadname,'(',loading,') = opusdde02('',name,'',"AB");']);
end;

clear filepath filename PLSext loading dot0 lnum loadname name;
```

APPENDIX A-4. OPUSDDE02.M

```
% opusdde02 extracts data blocks from Bruker's OPUS FT-IR software
% Written by Bruce T. Thompson, 01/23/2003
function ddedata = opusdde02(pathfilename,block);

% pathfilename contains the complete path\filename.ext
%block must be one of the following
%   "AB", "AB/Multiple", "Trace/Multiple", "AB/Integ","AB/Quant", "IgRf",
%   "IgSm", "ScRf", "AB/Peak"
%initialize dde connection with OPUS

channel = ddeinit('OPUS','OPUS|System');

% Initial command string components
rdfile = 'READ_FROM_FILE ';
rdblock = 'READ_FROM_BLOCK ';
rddata = 'READ_DATA';
matinfo = 'MATRIX_INFO';
matelem = 'MATRIX_ELEMENT';
repinfo = 'REPORT_INFO';
creturn = sprintf('\n');
tmult = 'Trace/Multiple';
ab = 'AB';
abmult = 'AB/Multiple';
intg = 'AB/Integ';
quant = 'AB/Quant';
igrf = 'IgRf';
igsm = 'IgSm';
scrf = 'ScRf';
peak = 'AB/Peak';
unfile = 'UNLOAD_FILE ';

% Load file
s = [rdfile,pathfilename,creturn];
response = ddereq(channel,s,[1,1]);

% file number is third returned value from read file
[okay,errstr] = isok(response);
if okay
   [fileno,rem] = strtok(response,creturn);
   [fileno,rem] = strtok(rem,creturn);
```

```
   fileno = strtok(rem,creturn);
else
   error('Error in reading file!\nError message:  %s',errstr);
end;


% Use the following line if ddereq text parameter set to [1,0]
%fileno = response(find(response~=0));

switch block
   case ab
                  % specify ab as the block to read
                  s = [rdblock,block,creturn];
                  response = ddereq(channel,s,[1,1]);

                  % read data
                  s = [rddata,creturn];
                  ddedata = ddereq(channel,s,[1,0]);
                  ddedata = ab2struc(ddedata);
   case igrf
                  % specify ab as the block to read
                  s = [rdblock,block,creturn];
                  response = ddereq(channel,s,[1,1]);

                  % read data
                  s = [rddata,creturn];
                  ddedata = ddereq(channel,s,[1,0]);
                  ddedata = ab2struc(ddedata);
   case intg
      %specify integration report as block
                  s = [rdblock,block,creturn];
                  response = ddereq(channel,s,[1,1]);
                  [okay,errstr] = isok(response);
                  if ~okay
                     error('Error specifying integration block!\nError message:  %s',errstr);
                  end;

      %read number of main and sub reports
      s = [repinfo,creturn];
      response = ddereq(channel,s,[1,1]);
                  [okay,errstr] = isok(response);
                  if ~okay
                    error('Error in reading report info!\nError message:  %s',errstr);
                  end;
```

```
[str,rem] = strtok(response,creturn);
[main,rem] = strtok(rem,creturn);
sub = strtok(rem,creturn);

k = 0;
for rept = 1:str2num(main)
   srept = num2str(rept,'%1d');
   %read matrix infor
   s = [matinfo,' ',srept,' ',sub,creturn];
   response = ddereq(channel,s,[1,1]);
                  [okay,errstr] = isok(response);
                  if ~okay
      error('Error in reading matrix info!\nError message:  %s',errstr);
                  end;
   response = ddereq(channel,s,[1,0]);
   rows = response(2);
   cols = response(3);

   % read matrix elements - no error checking
   for r = 1:rows
      rstr = num2str(r,'%1d');
      k = k + 1;

      % integration label
      s = [matelem,' ',srept,' ',sub,' ',rstr,' 1',creturn];
      response = ddereq(channel,s,[1,1]);
      [str,rem] = strtok(response,creturn);
      ddedata(k).label = strtok(rem,creturn);

      % integration type
      s = [matelem,' ',srept,' ',sub,' ',rstr,' 2',creturn];
      response = ddereq(channel,s,[1,1]);
      [str,rem] = strtok(response,creturn);
      ddedata(k).type = strtok(rem,creturn);

      %integration range
      s = [matelem,' ',srept,' ',sub,' ',rstr,' 4',creturn];
      response = ddereq(channel,s,[1,0]);
      range(1) = response(2);
      s = [matelem,' ',srept,' ',sub,' ',rstr,' 5',creturn];
      response = ddereq(channel,s,[1,0]);
      range(2) = response(2);
      ddedata(k).range = range;
```

```
            %integration value
            s = [matelem,' ',srept,' ',sub,' ',rstr,' 3',creturn];
            response = ddereq(channel,s,[1,0]);
            ddedata(k).value = response(2);
        end;
    end;

case peak
    %specify peak pick report as block
                s = [rdblock,block,creturn];
                response = ddereq(channel,s,[1,1]);
                [okay,errstr] = isok(response);
                if ~okay
        error('Error specifying peak table block!\nError message:  %s',errstr);
                end;
    %read number of main and sub reports
    s = [repinfo,creturn];
    response = ddereq(channel,s,[1,1]);
                [okay,errstr] = isok(response);
                if ~okay
        error('Error in reading report info!\nError message:  %s',errstr);
                end;
    [str,rem] = strtok(response,creturn);
    [main,rem] = strtok(rem,creturn);
    sub = strtok(rem,creturn);

    for rept = 1:str2num(main)
        srept = num2str(rept,'%1d');
        %read matrix infor
        s = [matinfo,' ',srept,' ',sub,creturn];
        response = ddereq(channel,s,[1,1]);
                        [okay,errstr] = isok(response);
                        if ~okay
            error('Error in reading matrix info!\nError message:  %s',errstr);
                        end;
        response = ddereq(channel,s,[1,0]);
        rows = response(2);
        cols = response(3);

        % read matrix elements - no error checking
        reportmat = zeros(rows,cols);
        for r = 1:rows
            for c = 1:cols
                rstr = num2str(r,'%1d');
```

```
            cstr = num2str(c,'%1d');
            s = [matelem,' ',srept,' ',sub,' ',rstr,' ',cstr,creturn];
            response = ddereq(channel,s,[1,0]);
            reportmat(r,c) = response(2);
         end;
      end;

      % wavenumbers
      ddedata(rept).wn  = reportmat(:,1);

      %intensity data
      ddedata(rept).absint  = reportmat(:,2);
      ddedata(rept).relint  = reportmat(:,3);

      %peak width
      ddedata(rept).pkwidth = reportmat(:,4);
   end;

case quant
   %specify quantitation report as block
            s = [rdblock,block,creturn];
            response = ddereq(channel,s,[1,1]);
            [okay,errstr] = isok(response);
            if ~okay
      error('Error specifying integration block!\nError message:  %s',errstr);
            end;

   %read number of main and sub reports
   s = [repinfo,creturn];
   response = ddereq(channel,s,[1,1]);
            [okay,errstr] = isok(response);
            if ~okay
      error('Error in reading report info!\nError message:  %s',errstr);
            end;
   [str,rem] = strtok(response,creturn);
   [main,rem] = strtok(rem,creturn);
   sub = strtok(rem,creturn);

   %read matrix infor
   s = [matinfo,creturn];
   response = ddereq(channel,s,[1,1]);
            [okay,errstr] = isok(response);
            if ~okay
      error('Error in reading matrix info!\nError message:  %s',errstr);
```

```
            end;
    response = ddereq(channel,s,[1,0]);
    rows = response(2);
    cols = response(3);

    % read matrix elements - no error checking
    nmain = str2num(main);
    n = 0;
    for m = 1:nmain
       main = num2str(m,'%1d');
       for r = 1:rows
          rstr = num2str(r,'%1d');

          % quantitation label
          s = [matelem,' ',main,' ',sub,' ',rstr,' 1',creturn];
          response = ddereq(channel,s,[1,1]);
                [okay,errstr] = isok(response);
          if okay
             n = n + 1;
             [str,rem] = strtok(response,creturn);
             stype = strtok(rem,creturn);
             ddedata(n).type = stype;

             %quantitation value
             s = [matelem,' ',main,' ',sub,' ',rstr,' 2',creturn];
             response = ddereq(channel,s,[1,0]);
             ddedata(n).value = response(2);
          end;
       end;
    end;

case tmult
            % specify Trace/Multiple as the block to read
            s = [rdblock,block,creturn];
            response = ddereq(channel,s,[1,1]);

            % read data
            s = [rddata,creturn];
            ddedata = ddereq(channel,s,[1,0]);

            % extract matrix from returned data
            nrows = ddedata(2)+1;
            ncols = ddedata(5);
            maxrow = size(ddedata,1)-1;
```

```
                ddedata = ddedata(8:maxrow);
                ddedata = reshape(ddedata,nrows,ncols);
                ddedata = ddedata(2:nrows,:);

   otherwise
      error('Block type incorrect or not implemented!\nBlock type:  %s',block);
end;

% Unload file
s = [unfile,fileno,creturn];
response = ddereq(channel,s,[1,1]);

%disconnect from OPUS
ddeterm(channel);
```

APPENDIX A-5. ISOK.M

```
function [logicout,errorstr] = isok(responsestr);

creturn = sprintf('\n');
logicout = 0;
ok = strtok(responsestr,creturn);
if strcmp(ok,'OK')
        logicout = 1;
end;
errorstr = ok;
```

APPENDIX A-6. ANALCONV04.M

```
% conversion function of analyzer a/d to ppm
% outfile = analconv(infile);
% converts column 3 voltage to NO [pppm] by multiplying by range specified in NOrange and
% places in column 8
% converts col 4 voltage to CO [ppm] by multiplying by 10000ppm/1V and
% places in column 9
% computes mg CO and puts the results in column 10 based on flow rate
% recorded in column 7
% computes mg NO and puts the results in column 11 based on flow rate
% recorded in column 7
% computes main and HW flow rates into columns 12 and 13
% smooths ppm traces based on window size supplied in wsize for assigning
% concentrations to FT-IR spectra (i.e. calibration standards)

function outfile = analconv04(infile,NOrange,wsize);

F = infile(:,6)/60;
MWNO = 30.01;
MWCO = 28.01;
% molar volume at 25C and 1atm
%Vm = 24.465;
% molar volume at 70F (23.9C) and 1atm
Vm = 24.3729525;
range = [1,3,10,30,100];
[r,c] = size(infile);

% dt = mean(infile(2:r,1)-infile(1:r-1,1));
dt = zeros(r,1);
dt(1) = infile(2,1)-infile(1,1);
dt(2:r) = infile(2:r,1)-infile(1:r-1,1);
temp = infile;
temp(:,c+1) = infile(:,3)*range(NOrange);
temp(:,c+2) = infile(:,4)*10000;
%temp(find(temp(:,c+1)<0),c+1)=0;
%temp(find(temp(:,c+2)<0),c+2)=0;
temp(:,c+3) = temp(:,c+1).*dt.*F*MWNO/(1000*Vm);
temp(:,c+4) = temp(:,c+2).*dt.*F*MWCO/(1000*Vm);
temp(:,c+5) = temp(:,6);
temp(:,c+6) = temp(:,7)/10;
```

```
temp(:,c+7) = filter(ones(1,wsize)/wsize,1,temp(:,c+1));
temp(:,c+8) = filter(ones(1,wsize)/wsize,1,temp(:,c+2));

outfile(1).time = temp(:,1);
outfile(1).event = temp(:,2);
outfile(1).NOvolts = temp(:,3);
outfile(1).COvolts = temp(:,4);
outfile(1).CO2volts = temp(:,5);
outfile(1).MainFlowvolts = temp(:,6);
outfile(1).HWFlowvolts = temp(:,7);
outfile(1).NOppm = temp(:,8);
outfile(1).COppm = temp(:,9);
outfile(1).NOmg = temp(:,10);
outfile(1).COmg = temp(:,11);
outfile(1).MainFlowLpm = temp(:,12);
outfile(1).HWFlowLpm = temp(:,13);
outfile(1).NOppmf = temp(:,14);
outfile(1).COppmf = temp(:,15);
```

APPENDIX A-7. COMPSTAT4.M

```
%compute stats
%compute stats of various columns of the data
%  Written by Bruce T. Thompson, 10/14/2003
%compstat2 - added manual calculation of tstat from differences,
%         - BTT, 10/16/2003
%compstat3 - added results from ttest and RMSEP
%         - BTT, 11/18/2003
%compstat4 - reduced output to just results from ttest of two data sets and
%           RMSEP
%         - added norm stat
% **This applies to data sets comparing to only one other set of results**
%         - BTT, 2/7/2004

function statsout = compstats4(data,compcols);

[drows,dcols] = size(data);
[crows,ccols] = size(compcols);
if min(compcols)<0
   error('Comparison column(s) is smaller than matrix dimension');
end;
if max(compcols)>dcols
   error('Comparison column(s) is larger than matrix dimension');
end;
if (ccols~=dcols)
   error('Number of comparison columns is not the same size as data columns');
end;

statsout = zeros(8,dcols);
for i = 1:ccols
   if compcols(i)>0
      [h,significance,ci,statsi] = ttest2(data(:,i),data(:,compcols(i)),0.05);
      diffs = data(:,i)-data(:,compcols(i));
      [rdiffs,cdiffs] = size(diffs);
      [h,sig,cis,sst] = ttest(data(:,i),data(:,compcols(i)),0.05,'both');
      statsout(1,i) = h;
      statsout(2,i) = abs(sst.tstat);
      statsout(3,i) = sig;
      statsout(4,i) = sst.df;
      statsout(5,i) = mean(cis);
      statsout(6,i) = sst.sd;
```

```
        statsout(7,i) = sqrt((diffs'*diffs)/rdiffs);
    end;
end;
ccols = find(compcols~=0);
nstat(1,:) = statsout(5,ccols)/norm(statsout(5,ccols));
nstat(2,:) = statsout(6,ccols)/norm(statsout(6,ccols));
statsout(8,ccols) = sqrt(sum(nstat.^2));
```

APPENDIX A-8. TAUCALC.M

```
function taus = taucalc(X,wsize)

% taucalc calculates an arrays of time constants (slopes) for window sizes min(wsize)
% to max(wsize), odd.  If wsize is not specified, wsize = [3,5,7,9,11]
% for the signal X(time,amplitude) (amplitude = intensity, concentration or absorbance)
% Only the first two columns of X are used and time is column 1

% error checking
if nargin<2
   wsize = [3,5,7,9,11];
else
   wsize = ((fix(wsize/2)+0.5)*2);
end;
[rwsize,cwsize] = size(wsize);
if (rwsize~=1 & cwsize~=1)
   error('Window size not a vector');
end;
[xrows,xcols] = size(X);
if xcols<2
   error('Input array has insufficient columns');
end;
colors = ['k';'r';'g';'b';'m';'k';'r';'g';'b';'m'];

% norm = norm(data)
dt = mean(X(3:xrows,1)-X(2:xrows-1,1));
dp = fix(xrows/2);

% raise baseline to just above zero to remove negative numbers
X(:,2) = norm(X(:,2))
X(:,2) = X(:,2)-min(X(:,2));
X(:,2) = log(X(:,2));
nonos = find(normlog(:,2)==-Inf);
if ~isempty(nonos)
          [nrows,ncols] = size(nonos);
          for i = 1:nrows
      if nonos(i)==1
         normlog(1,2) = normlog(2,2);
      elseif nonos(i)==xrows
         normlog(nonos(i),2) = normlog(nonos(i)-1,2);
      else
```

```
            normlog(nonos(i),2) = mean([normlog(nonos(i)-1,2),normlog(nonos(i)+1,2)]);
        end;
            end;
end;
plot(X(:,1),normlog(:,2));
hold on;

for i = 1:max([rwsize,cwsize])
    taus(i,1) = wsize(i);
    derivdata = savgol(normlog(:,2)',wsize(i),2,1);
    taus(i,2) = (-1/min(derivdata(dp:xrows)))/dt;
    taus(i,3) = find(derivdata==min(derivdata(dp:xrows)));
    plot(X(:,1),derivdata,colors(i,:));
%    if i==1 hold on; end;
end;
hold off;
```

APPENDIX A-9. FINDCOORDS3.M

```
function xyz = findcoords3(xyz,K,nd);

% findcoords finds the numbers of the array element that is closest to x and z
% if X is an array, it only looks at the first row and only the first
% column of Z. this is to conform to oapms
% xyz is assumed to be a three element vector

x = (xyz(1)-X(1,:)).^2;
z = (xyz(3)-Z(:,1)).^2;
x = find(x==min(x));
xyz(1) = x(1);
z = find(z==min(z));
xyz(3) = z(1);
```

APPENDIX A-10. HW55CMWCA.M

```
% HWGc - script to simulate photons going through a hollow waveguide fiber
%  version a is basic 2D without any losses - it is designed to get a working
%  framework for the HWG.
%
% Axis coodination:  photons from +x infinity are reflected onto the yaxis
%  through the focal point at (f,0) into the HWG at (f,ys).  The source mirror
%  is always at -y coordinates (so the focal point is an the xaxis.
%
%  B. T. Thompson, 12/21/01
% hwgf - "inside" path. Labeled plot.  BTT 1/27/02
% hwgg - "outside" path.  BTT 1/29/02
% hwgg55cmf - 55cm waveguide and varies mirror 1 and 2
% hwgg55cmg - 55cm waveguide and varies mirror 1 in x and y directions
% hw55cmwca - 55cm waveguide and Winston cone on detector 4/6/2004
% run simulation on two parameters first: where the wc opening is and how
% big.  The simulate xy position of mirror at optimum wc size and position

%_____
____
% set up physical constants
pixels = 1025;  %number of pixels along y axis
pathlength = 550;   %HWG pathlength in mm
hwgid = 2.0;    %id of HWG in mm
f = 25.4;   %focal length of mirror
plotphoton = 0;
plotnormal = 0;
savedata = 0;
bydiff = 1;
dontskip = 1;

%_____
____
%header
fname = 'hw55wca01';
if savedata fid = fopen([fname,'.txt'],'w'); end;
clc;
disp(' ')
disp('_____
_____')
disp(' No.   angle?     a       WC x     No.Photons   Throughput   Avg Dist      Time    ')
```

```
disp('_____
_____')
sout = sprintf('  No.   angle?     a          WC x     No.Photons   Throughput   Avg Dist
Time\n');

%_____
____
%Setup figure
if savedata fwrite(fid,sout); end;
if plotphoton
        copper = [1,0.62,0.40];
        silver = [0.684,0.684,0.684];
   figure(4);
   set(gcf,'Position',[50 250 950 450]);
   ttext = {'Mirror 1','Winston Cone','Hollow Waveguide','Detector'};
   x = zeros(4,1);
   y = zeros(4,1);
end;


%_____
____
% set up variables
cas = 50.8; %cas is the size of the off axis parabolic mirror (oapm), source end
pixelres = cas/pixels;  %pixel resolution

% mirror arrays have four elements.  Elements 1 & 2 are the x & y coordinates.
% Elements 3 & 4. Element 5 is the angle between then normal ray and the x-axis
oapms = zeros(pixels,5);    % off-axis parabolic mirror coordinates - source end
nf = 1; %focal length multiplier to investigating different entrance optics
fs = nf*f; %distance on yaxis where the focal point is
ds = 2*fs;  %ds is the distance between optical and mechanical axes on source end
hwgboundary = hwgid/2; %HWG boundaries above and below HWG axis, parallel to the
yaxis.
hwgup = fs + hwgboundary;
hwgdn = fs - hwgboundary;
minlength = pathlength;
maxlength = 2*minlength;
fd = f;
fd2 = fd;
dd = 2*fd;
xaxis = [1;0];
yaxis = [0;1];
m1x = 0;
```

```
m1y = 0;
m2x = 2;
m2y = f;
detx = 0;  %detector distance for end of HWG in mm
dety = 0;
m15 = tan(15/180);
runnumber = 0;
ckangle = 0;
nphotons = 10;


%_____
____
%oapms is y and x coordinates of the mirror surface and the slope of the line
%connecting the mirror to the focus.
% Mirror1 focusses from the source into the HWG
% Mirror2 collects from the HWG and projects to Mirror3
% Mirror3 collects from Mirror2 and focusses on the detector
%In this scheme, the mirror focal point is always (0,fs)
for i = 1:pixels
% Calculate the first mirror (entrance to HW) location in millimeters
   oapms(i,2) = (ds + (pixels/2 - i)*pixelres);
   oapms(i,1) = (oapms(i,2).^2)/(4*fs);

%Normal vector from zero
   oapms(i,3) = 1;
   oapms(i,4) = -(oapms(i,2)/(2*fs));
   magn = norm(oapms(i,3:4));
   oapms(i,3:4) = oapms(i,3:4)/magn;
   oapms(i,5) = acos(oapms(i,3:4)*xaxis);

% Rotate the Mirror1 pi/2 counterclockwise
   phi = pi/2;
   transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
   oapms(i,1:2) = (transmat*oapms(i,1:2)')';
   oapms(i,3:4) = (transmat*oapms(i,3:4)')';
end;

 runnumber = 0;
 for ckangle = 0:1

%_____
____
% Calculate Winston Cone
```

```
% for a = 1:0.1:5
a = 2.005; %halfsize of opening
ap = 0.405; %halfsize of exit
theta = asin(ap/a); %half angle of opening
wclength = (a+ap)*cot(theta); %length of cone
wcpixels = fix(wclength/pixelres);  %pixel resolution as defined in oapmconfig, for consistency
wcf = ap*(1+sin(theta));  %focal length of Winston cone

% Variables for varying cone curvature
i = 1:wcpixels;
ares = (a - ap)/wcpixels;
api = a - i*ares;
thetai = asin(api/a);
fi1 = api.*(1+sin(thetai));
winston = zeros(wcpixels,5);    % winston cone coordinates

% if runnumber<100 runnumber = runnumber + 1; else return; end;

for i = 1:wcpixels
%     disp(i)
          % Calculate the mirror location in (r',z') coordinates
          winston(i,1) = (i*pixelres);
          winston(i,2) = sqrt((winston(i,1)+wcf)*(4*wcf));

   %Normal vector from zero
          winston(i,3) = 1;
          winston(i,4) = -(winston(i,2)/(2*wcf));
          winston(i,3:4) = winston(i,3:4)/norm(winston(i,3:4));
          winston(i,5) = acos(winston(i,3:4)*xaxis);
          if (winston(i,4)<0)&(winston(i,3)>0) winston(i,5) = -winston(i,5); end;

          % Rotate theta clockwise
          transmat = [cos(-theta) -sin(-theta);sin(-theta) cos(-theta)];
          winston(i,1:2) = (transmat*winston(i,1:2)')';
          winston(i,3:4) = (transmat*winston(i,3:4)')';
          winston(i,5) = acos(winston(i,3:4)*xaxis);
          if (winston(i,4)<0)&(winston(i,3)>0) winston(i,5) = -winston(i,5); end;
end;    %for generating winston cone curve

upwinston = winston;
upwinston(:,1) = -1*upwinston(:,1);
upwinston(:,3) = -1*upwinston(:,3);
upwinston(:,5) = acos(upwinston(:,3:4)*xaxis);
```

```
%reflect upwinston across x-axis to dnwinston
dnwinston = upwinston;
dnwinston(:,2) = -1*dnwinston(:,2);
dnwinston(:,4) = -1*dnwinston(:,4);
dnwinston(:,5) = acos(dnwinston(:,3:4)*xaxis);

upwinstonorig = upwinston;
dnwinstonorig = dnwinston;

%_____
____
%Place mirror and detector
mirror1 = oapms;
detcoords = [0 0 1 2];  %Detector coordinates in [left bottom xdimension ydimension]
dcenter = detcoords(4)/2;
deltad = (maxlength - 0)/pixels;


%_____
____
%beam profile initialization
% bpsize = 50;
%beamres = (hwgup - hwgdn)/pixels;
% beamres = bpsize/pixels;


%_____
____
% Begin looping
%       for m1x = -5:0.25:5
                hwgs = m1x;   %hwgs is the point on the xaxis where the HWG starts
                hwgd = hwgs+minlength;    %where the HWG ends, detector end
%       for m1y = -1.1:0.1:1.1   %xd refers to the verticle position of mirror 1

        %Begin loop to move mirrors 2
        for m2x = 0:0.5:20
%           for m2y = 0:20
                        %x,y translate values for mirrors 1, 2, 3, and detector
                        optrans = zeros(4,2);
                        optrans(1,:) = [m1x, m1y];
                        optrans(2,:) = [hwgd+wclength+m2x, m2y];
        optrans(3,:) = [optrans(2,1)+detx,optrans(2,2)-dcenter+dety];
        tic;
        phi = pi;
```

```
        %Translate the Mirror1 so the focal point is m1x from the end of the HWG
         mirror1(:,1) = oapms(:,1) + optrans(1,1);
         mirror1(:,2) = oapms(:,2) + optrans(1,2);
    %Translate the WC to the end of the HW
        upwinston(:,1) = upwinstonorig(:,1) + optrans(2,1);
        upwinston(:,2) = upwinstonorig(:,2) + optrans(2,2);
        dnwinston(:,1) = dnwinstonorig(:,1) + optrans(2,1);
        dnwinston(:,2) = dnwinstonorig(:,2) + optrans(2,2);
        %Translate detector to focal point of Mirror3
        optrans(3,1) = upwinston(1,1);
        detcoords(1:2) = optrans(3,:);
        pdet = repmat(detcoords(1:2),5,1);
        pdet(2,2) = pdet(2,2) + detcoords(4);
        pdet(3,:) = pdet(3,:) + detcoords(3:4);
        pdet(4,1) = pdet(4,1) + detcoords(3);

        %Beam profile placement
        runnumber = runnumber + 1;
%           beamprofile = zeros(pixels,2500);
%           bpx = hwgd+10;
%           botbpy = fs-bpsize/2;

        %Plot the physical layout of the optics
        if plotphoton
           figure(4);
           plot(mirror1(:,1),mirror1(:,2),'-','color',copper,'linewidth',2);
           hold on;
                                plot(upwinston(:,1),upwinston(:,2),'-k');
                                plot(dnwinston(:,1),dnwinston(:,2),'-k');
           plot([hwgs hwgd],[hwgup hwgup],'Color',silver);
           plot([hwgs hwgd],[hwgdn hwgdn],'Color',silver);
           plot([hwgs hwgs],[hwgup hwgdn],'Color',silver);
           plot([hwgd hwgd],[hwgup hwgdn],'Color',silver);
           plot(pdet(:,1),pdet(:,2));
           xlabel('x dimension (mm)');
           ylabel('y dimension (mm)');
           title('Spacial Setup of Hollow Waveguide Simulation');
           xlim([mirror1(1,1)-75, detcoords(1,1)+75]);
           ylim([mirror1(1025,2)-75, mirror1(1,2)+75]);
           set(gca,'PlotBoxAspectRatioMode','manual','PlotBoxAspectRatio',[1     150/450
1]);
           %set coordinates for Mirror 1 label
           x(1) = mirror1(512,1)-40;
           y(1) = mirror1(512,2)-20;
```

```
    %set coordinates for WC label
    x(2) = optrans(2,1)-40;
    y(2) = optrans(2,2)-10;
    %set coordinates for Hollow Waveguide label
    x(3) = hwgd/2-50;
    y(3) = hwgup + 10;
    %set coordinates for Detector label
    x(4) = optrans(3,1)+5;
    y(4) = optrans(3,2)+2;
    text(x,y,ttext);
%           drawnow;
    hold off;
end;

photoncount = 0;
sumdist = 0;
photondist = zeros(pixels,1);   %distribution of photon distances
halfbeam = fix(20.0/pixelres);
sourcebeam = 513-halfbeam:513+halfbeam;
beam = 2*halfbeam+1;
mirrordiff = zeros(pixels,1);
mirror3proj = zeros(100,2);
hitmirror3 = 0;
pinctotal = 0;
unilaunch = 1;
if unilaunch
    nphotons = beam;
    startpt = 1:nphotons;
%           startpt = 1:fix(beam/nphotons):nphotons*(fix(beam/nphotons));
end;

missed = zeros(nphotons,1);
mnumber = 0;
for i = 1:nphotons
    %Launch photon
    hitdetect = 0;
    photon = zeros(1024,2);
    if unilaunch
        photonstart = sourcebeam(startpt(i));
    else
        photonstart = sourcebeam(fix(rand(1)*beam)+1);
    end;
    photon(1,:) = [mirror1(photonstart,1) mirror1(1,2)+100];
```

```
%locate where ray hits mirror1, translate to origin and normalize
nray = zeros(2,2);
photon(2,:) = mirror1(photonstart,1:2);

%calculate the reflected ray
phiw = 0.5*pi;
if bydiff
   phiv = -(2*mirror1(photonstart,5)-phiw);
   nray(2,:) = [cos(phiv),sin(phiv)];
else
   phi = 2*(phiw-mirror1(photonstart,5));
   transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
   nray(2,:) = (transmat*nray(1,:)')';
end;


m = nray(2,2)/nray(2,1);
if  ckangle & abs(m)>m15 continue; end;
pinc = 2;
%Check to make sure photon hits hwg hole
photon(pinc+1,1) = hwgs;
photon(pinc+1,2) = ((photon(pinc+1,1)-photon(pinc,1))*m)+photon(pinc,2);

if (photon(pinc+1,2)>=hwgdn)&(photon(pinc+1,2)<=hwgup)
   pinc = pinc+1;
   while (photon(pinc,1)<hwgd);
      pinc = pinc+1;
      if m>0
         photon(pinc,2) = hwgup;
         photon(pinc,1)  =  ((photon(pinc,2)-photon(pinc-1,2))/m)+photon(pinc-
1,1);
         m = -m;
      elseif m<0
         photon(pinc,2) = hwgdn;
         photon(pinc,1)  =  ((photon(pinc,2)-photon(pinc-1,2))/m)+photon(pinc-
1,1);
         m = -m;
      else
         photon(pinc,2) = photon(pinc-1,2);
         photon(pinc,1) = hwgd+1;
      end;
   end;
   m = -m;
   if m~=0
```

```
                photon(pinc,1) = hwgd;
                photon(pinc,2)    =    ((photon(pinc,1)-photon(pinc-1,1))*m)+photon(pinc-
1,2);
            else
                photon(pinc,1) = hwgd;
                photon(pinc,2) = photon(pinc,2);
            end;
%               stopphoton = pinc;
%                bpinc = round(((((bpx-photon(stopphoton,1))*m)+photon(stopphoton,2))-
botbpy)/beamres);
%                 beamprofile(bpinc,runnumber) = beamprofile(bpinc,runnumber) + 1;

            phdist = 0;
            for n = 3:pinc-1
                phdist = phdist + pdist(photon(n:n+1,:));
            end;

            pinc_1 = pinc;
            pinc = pinc + 1;
                                updown = 0;
                                if m>0 updown = 1; end;
                                if m<0 updown = -1; end;
                                if (m~=0)
                                        goodray = 1;
                                        loopcheck = 0;
                                        while
(photon(pinc_1,1)<upwinston(1,1))&goodray
                loopcheck = loopcheck + 1;

                                                % Does the photon go straight
through the exit aperture?
                mirrordiff = ((upwinston(1,1)-photon(pinc_1,1))*m)+photon(pinc_1,2);
                if ((mirrordiff(1)<=upwinston(1,2))&(mirrordiff(1)>=dnwinston(1,2)))
                    photon(pinc,1) = upwinston(1,1);
                    photon(pinc,2)                =                ((photon(pinc,1)-
photon(pinc_1,1))*m)+photon(pinc_1,2);
                    pinc_1 = pinc;
                    pinc = pinc + 1;

                                                % If the m>0, the photon will hit the
upper side of the Winston cone
                elseif (updown>0&m>0)
                    mirrordiff                =                ((upwinston(:,1)-
photon(pinc_1,1))*m)+photon(pinc_1,2);
```

```
                    if
((mirrordiff(wcpixels)<=upwinston(wcpixels,2))&(mirrordiff(1)>=upwinston(1,2)))
                        mirrordiff = (upwinston(:,2)-mirrordiff).^2;
                        mirrorpt = find(mirrordiff==min(mirrordiff));
                        nray(1,:) = (photon(pinc_1,1:2)-upwinston(mirrorpt,1:2));
                        nray(1,:) = nray(1,:)/norm(nray(1,:));

                        photon(pinc,:) = upwinston(mirrorpt,1:2);
                        pinc_1 = pinc;
                        pinc = pinc + 1;

                        %calculate the reflected ray
                        nray(2,:)                =                -(nray(1,:)                -
2*(nray(1,1:2)*upwinston(mirrorpt,3:4)')*upwinston(mirrorpt,3:4));
                        m = nray(2,2)/nray(2,1);
                        updown = -updown;
                    else
                        goodray = 0;
                    end;
                elseif (updown<0&m<0)
                    mirrordiff                =                ((dnwinston(:,1)-
photon(pinc_1,1))*m)+photon(pinc_1,2);
                    if
((mirrordiff(wcpixels)>=dnwinston(wcpixels,2))&(mirrordiff(1)<=dnwinston(1,2)))
                        mirrordiff = (dnwinston(:,2)-mirrordiff).^2;
                        mirrorpt = find(mirrordiff==min(mirrordiff));
                        nray(1,:) = (photon(pinc_1,1:2)-dnwinston(mirrorpt,1:2));
                        nray(1,:) = nray(1,:)/norm(nray(1,:));

                        photon(pinc,:) = dnwinston(mirrorpt,1:2);
                        pinc_1 = pinc;
                        pinc = pinc + 1;

                        %calculate the reflected ray
                        nray(2,:)                =                -(nray(1,:)                -
2*(nray(1,1:2)*dnwinston(mirrorpt,3:4)')*dnwinston(mirrorpt,3:4));
                        m = nray(2,2)/nray(2,1);
                        updown = -updown;
                    else
                        goodray = 0;
                    end;
                else
                    goodray = 0;
                end; %if ((mir
```

```
                    if loopcheck>1e3
                        error('Endless loop detected');
                    end;
                end;    %while
            end;    %(m~=0)
            %Project to detector
            photon(pinc,:)              =              [detcoords(1)              ((detcoords(1)-
photon(pinc_1,1))*m)+photon(pinc_1,2)];
            if
(photon(pinc,2)<=detcoords(2)+detcoords(4))&(photon(pinc,2)>=detcoords(2))
                sumdist = sumdist + phdist;
                n = floor((phdist-0)/deltad)+1;
                photondist(n) = photondist(n) + 1;
                photoncount = photoncount + 1;
                hitdetect = 1;
            end;
        end;    %if hit hole

        %Plot photon trace
        if plotphoton
            figure(4);
            hold on;
            for k = 1:pinc-1
                plot(photon(k:k+1,1),photon(k:k+1,2),'r');
            end;
            hold off;
        end;

        pinctotal = pinctotal + pinc-3;
        %plot normal vectors at mirrors
        if plotnormal&plotphoton
            hold on;

    plot([mirror1(photonstart,3)*10+mirror1(photonstart,1)...
            mirror1(photonstart,1)],...
            [mirror1(photonstart,4)*10+mirror1(photonstart,2) ...
            mirror1(photonstart,2)],'-k');

    plot([mirror2(mirrorpt2,3)*10+mirror2(mirrorpt2,1)...
            mirror2(mirrorpt2,1)],...
            [mirror2(mirrorpt2,4)*10+mirror2(mirrorpt2,2) ...
            mirror2(mirrorpt2,2)],'-k');

    plot([mirror3(mirrorpt3,3)*10+mirror3(mirrorpt3,1)...
```

```
                mirror3(mirrorpt3,1)],...
                [mirror3(mirrorpt3,4)*10+mirror3(mirrorpt3,2) ...
                mirror3(mirrorpt3,2)],'-k');
                                        hold off;
            end; %if plotnormal
        end;    %for i = photon number
        %       sout = sprintf('Elapsed time:  %6.2f',toc/60);
        %       disp(sout);
        if plotphoton
                xlim([-100 650]);
                ylim([0 75]);
           drawnow;
        end;
        if photoncount>0 avgdist = sumdist/photoncount;
           else avgdist = 0; end;
        sout = sprintf('%4d      %1d      %6.2f     %6.2f      %6d          %6.2f      %6.1f
%6.2f',...

runnumber,ckangle,a,m2x,photoncount,photoncount*100/nphotons,avgdist,toc/60);
        disp(sout);
        if savedata
           sout = sprintf('%4d      %1d      %6.2f     %6.2f      %6d          %6.2f      %6.1f
%6.2f\n',...
               runnumber,ckangle,a,m2x,photoncount,avgdist,toc/60);
           fwrite(fid,sout);
        end;
      end;    %for m2x =
   end;    %for a =
end;    %for ckangle

%_____________________________________________________________________
____
% tidy up
if savedata
   fclose(fid);
%     eval('save ',fname,'bpdata beamprofile');
end;
```

APPENDIX A-11. HWDET.M

```
%HWdet

Det.size = [0,1,1];
Det.coords = zeros(4,3);
Det.center = [550,f+150,0];
Det.coords(1,:) = Det.center + Det.size.*[1,-1,-1];
Det.coords(2,:) = Det.center + Det.size.*[1,-1,1];
Det.coords(3,:) = Det.center + Det.size.*[1,1,1];
Det.coords(4,:) = Det.center + Det.size.*[1,1,-1];
Det.N = [1,0,0];
Det.minmax = [min(Det.coords);max(Det.coords)];
```

APPENDIX A-12. HWDETWC.M

```
%HWdetwc

Det.size = [0,1,1];
Det.coords = zeros(4,3);
Det.center = [max(max(wc.X)),f,0];
Det.coords(1,:) = Det.center + Det.size.*[1,-1,-1];
Det.coords(2,:) = Det.center + Det.size.*[1,-1,1];
Det.coords(3,:) = Det.center + Det.size.*[1,1,1];
Det.coords(4,:) = Det.center + Det.size.*[1,1,-1];
Det.N = [-1,0,0];
Det.minmax = [min(Det.coords);max(Det.coords)];
```

APPENDIX A-13. HWGG55CMG.M

```
% HWGc - script to simulate photons going through a hollow waveguide fiber
%  version a is basic 2D without any losses - it is designed to get a working
%  framework for the HWG.
%
% Axis coodination:  photons from +x infinity are reflected onto the yaxis
%  through the focal point at (f,0) into the HWG at (f,ys).  The source mirror
%  is always at -y coordinates (so the focal point is an the xaxis.
%
%  B. T. Thompson, 12/21/01
% hwgf - "inside" path. Labeled plot.  BTT 1/27/02
% hwgg - "outside" path.  BTT 1/29/02
% hwgg55cmf - 55cm waveguide and varies mirror 1 and 2
% hwgg55cmg - 55cm waveguide and varies mirror 1 in x and y directions

% set up physical constants
pixels = 1025;  %number of pixels along y axis
pathlength = 550;   %HWG pathlength in mm
hwgid = 2.0;    %id of HWG in mm
f = 25.4;   %focal length of mirror
plotphoton = 0;
plotnormal = 0;
savedata = 0;
bydiff = 1;
dontskip = 1;

%header
fname = 'hwg55g04b';
if savedata fid = fopen([fname,'.txt'],'w'); end;
clc;
disp(' ')
disp('_____
_____')
disp(' No.    Mirror 1x       Mirror 1y      No.Photons    Throughput    Avg Dist       Time
')
disp('_____
_____')
sout = sprintf('  No.    Mirror 1x       Mirror 1y      No.Photons    Throughput    Avg Dist
Time\n');
if savedata fwrite(fid,sout); end;
if plotphoton
```

```
    figure(4);
    set(gcf,'Position',[406   498   950   450]);
    ttext = {'Mirror 1','Mirror 2','Mirror 3','Hollow Waveguide','Detector'};
    x = [  -73.2639;550.55;475.45;149.06;615.62];
    y = [   -11.95;4.71;109.97;40.53;91.12];
end;


% set up variables
cas = 50.8; %cas is the size of the off axis parabolic mirror (oapm), source end
pixelres = cas/pixels;  %pixel resolution
%circle to draw 101mm diameter detector housing
r = 101/2;
circumpix = round(r*2*pi/pixelres);
cirinc = r*2/(fix(circumpix/2)-1);
circle = zeros(circumpix,2);

circle(1:fix(circumpix/2),1) = (-50.5:cirinc:50.5)';
circle(fix(circumpix/2)+1:circumpix,1) = (50.5:-cirinc:-50.5)';
circle(1:fix(circumpix/2),2) = -sqrt(r^2-circle(1:fix(circumpix/2),1).^2);
circle(fix(circumpix/2)+1:circumpix,2) = -circle([fix(circumpix/2):-1:1],2);

% mirror arrays have four elements.  Elements 1 & 2 are the x & y coordinates.
% Elements 3 & 4. Element 5 is the angle between then normal ray and the x-axis
oapms = zeros(pixels,5);    % off-axis parabolic mirror coordinates - source end
oapmd = oapms;    % detector end of hwg
oapmd2 = oapms;
nf = 1; %focal length multiplier to investigating different entrance optics
fs = nf*f; %distance on yaxis where the focal point is
ds = 2*fs;  %ds is the distance between optical and mechanical axes on source end
hwgboundary = hwgid/2; %HWG boundaries above and below HWG axis, parallel to the
yaxis.
hwgup = fs + hwgboundary;
hwgdn = fs - hwgboundary;
minlength = pathlength;
maxlength = 2*minlength;
m1y = 0; %now on the detector side
fd = f;
fd2 = fd;
dd = 2*fd;
xaxis = [1;0];
yaxis = [0;1];
m1x = 0;
m1y = 0;
```

```
m2x = 0;
m2y = 0;
m3x = 0;
m3y = 190 + m2y;
detx = 0;
dety = 0;
m15 = tan(15/180);
nphotons = 1000;

%oapms is y and x coordinates of the mirror surface and the slope of the line
%connecting the mirror to the focus.
% Mirror1 focusses from the source into the HWG
% Mirror2 collects from the HWG and projects to Mirror3
% Mirror3 collects from Mirror2 and focusses on the detector
%In this scheme, the mirror focal point is always (0,fs)
for i = 1:pixels
% Calculate the first mirror (entrance to HW) location in millimeters
   oapms(i,2) = (ds + (pixels/2 - i)*pixelres);
   oapms(i,1) = (oapms(i,2).^2)/(4*fs);

%Normal vector from zero
   oapms(i,3) = 1;
   oapms(i,4) = -(oapms(i,2)/(2*fs));
   magn = norm(oapms(i,3:4));
   oapms(i,3:4) = oapms(i,3:4)/magn;
   oapms(i,5) = acos(oapms(i,3:4)*xaxis);

% Calculate the second (exit of HW) mirror location in millimeters
   oapmd(i,2) = (dd + (pixels/2 - i)*pixelres);
   oapmd(i,1) = (oapmd(i,2).^2)/(4*fd);

%Normal vector from zero
   oapmd(i,3) = 1;
   oapmd(i,4) = -(oapmd(i,2)/(2*fd));
   magn = norm(oapmd(i,3:4));
   oapmd(i,3:4) = oapmd(i,3:4)/magn;
   oapmd(i,5) = acos(oapmd(i,3:4)*xaxis);

%Duplicate for other two mirrors
%    oapmd(i,:) = oapms(i,:);
   oapmd2(i,:) = oapmd(i,:);

% Rotate the Mirror1 pi/2 counterclockwise
   phi = pi/2;
```

```
   transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
   oapms(i,1:2) = (transmat*oapms(i,1:2)')';
   oapms(i,3:4) = (transmat*oapms(i,3:4)')';

% Rotate the Mirror2 pi/2 counterclockwise and reflect through y-axis
   phi = pi*0.5;
   transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
   oapmd(i,1:2) = (transmat*oapmd(i,1:2)')';
   oapmd(i,3:4) = (transmat*oapmd(i,3:4)')';
   oapmd(i,1) = -oapmd(i,1);
   oapmd(i,3) = -oapmd(i,3);
   oapmd(i,5) = acos(oapmd(i,3:4)*xaxis);

% Rotate the Mirror3 pi counterclockwise
   phi = pi*0.5;
   transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
%    oapmd2(i,:) = oapmd(i,:);
   oapmd2(i,1:2) = (transmat*oapmd2(i,1:2)')';
   oapmd2(i,3:4) = (transmat*oapmd2(i,3:4)')';
   oapmd2(i,1) = -oapmd2(i,1);
   oapmd2(i,3) = -oapmd2(i,3);
   oapmd2(i,2) = -oapmd2(i,2);
   oapmd2(i,4) = -oapmd2(i,4);
   oapmd2(i,5) = -acos(oapmd2(i,3:4)*xaxis);


end;
mirror1 = oapms;
mirror2 = oapmd;
mirror3 = oapmd2;
detcoords = [0 0 -1 2];  %Detector coordinates in [left bottom xdimension ydimension]
dcenter = detcoords(4)/2;
deltad = (maxlength - 0)/pixels;
bpsize = 50;
%beamres = (hwgup - hwgdn)/pixels;
beamres = bpsize/pixels;
m1x = -2;
m1y = 0.1;
runnumber = 0;
ckangle = 1;
%  for ckangle = 0:1
%        for m1x = -5:0.25:5
                hwgs = 0;   %hwgs is the point on the xaxis where the HWG starts
                hwgd = hwgs+minlength;    %where the HWG ends, detector end
```

```
%       for m1y = -1.1:0.1:1.1   %xd refers to the verticle position of mirror 1

        %Begin loop to move mirrors 2
%       for m2x = -25:25
%          for m2y = -10:10
                        %x,y translate values for mirrors 1, 2, 3, and detector
                        optrans = zeros(4,2);
                        optrans(1,:) = [m1x, m1y];
                        optrans(2,:) = [hwgd+m2x, m2y];
        optrans(3,:) = [optrans(2,1)+m3x,optrans(2,2)+m3y];
        optrans(4,:) = [optrans(3,1)+detx,optrans(3,2)-f-dcenter+dety];
        tic;
        phi = pi;
        %Translate the Mirror1 so the focal point is m1x from the end of the HWG
         mirror1(:,1) = oapms(:,1) + optrans(1,1);
         mirror1(:,2) = oapms(:,2) + optrans(1,2);
        %Translate the Mirror2 so the focal point is m2x from the end of the HWG
        mirror2(:,1) = oapmd(:,1) + optrans(2,1);
        mirror2(:,2) = oapmd(:,2) + optrans(2,2);

        %Translate the Mirror3 to line up mechanical axis with detector1
        % mirror and be 100mm away from
        mirror3(:,1) = oapmd2(:,1) + optrans(3,1);
        mirror3(:,2) = oapmd2(:,2) + optrans(3,2);

        %Translate detector to focal point of Mirror3
        detcoords(1:2) = optrans(4,:);

        %Beam profile placement
        runnumber = runnumber + 1;
        beamprofile = zeros(pixels,2500);
        bpx = hwgd+10;
        botbpy = fs-bpsize/2;

        %Plot the physical layout of the optics
        if plotphoton
           figure(4);
           plot(mirror1(:,1),mirror1(:,2),'-','color',[0,0,0.6],'linewidth',2);
           hold on;
           plot(mirror2(:,1),mirror2(:,2),'-','color',[0.6,0,0],'linewidth',2);
           plot(mirror3(:,1),mirror3(:,2),'-','color',[0,0.5,0],'linewidth',2);
           plot([hwgs hwgd],[hwgup hwgup]);
           plot([hwgs hwgd],[hwgdn hwgdn]);
           plot([hwgs hwgs],[hwgup hwgdn]);
```

```
            plot([hwgd hwgd],[hwgup hwgdn]);
            circleplot(:,1) = circle(:,1)+optrans(3,1)-r;
            circleplot(:,2) = circle(:,2)+optrans(3,2)-f;
            plot(circleplot(:,1),circleplot(:,2),'r','linewidth',2);
            plot([detcoords(:,1) detcoords(:,1)+detcoords(:,3)],...
               [detcoords(:,2) detcoords(:,2)]);
            plot([detcoords(:,1) detcoords(:,1)+detcoords(:,3)],...
               [detcoords(:,2)+detcoords(:,4) detcoords(:,2)+detcoords(:,4)]);
            plot([detcoords(:,1) detcoords(:,1)],...
               [detcoords(:,2) detcoords(:,2)+detcoords(:,4)]);
            plot([detcoords(:,1)+detcoords(:,3) detcoords(:,1)+detcoords(:,3)],...
               [detcoords(:,2) detcoords(:,2)+detcoords(:,4)]);
         %          plot([bpx bpx],[botbpy-bpsize botbpy+bpsize],'-k');
         %          plot([-100 700],[fs fs]);
            xlabel('x dimension (mm)','FontName','Arial','FontSize',12);
            ylabel('y dimension (mm)','FontName','Arial','FontSize',12);
            title('Spacial Setup of Hollow Waveguide
Simulation','FontName','Arial','FontSize',12);
            xlim([mirror1(1,1)-25, mirror2(1,1)+25]);
            ylim([mirror1(1025,2)-75, mirror3(1025,2)+75]);
            set(gca,'PlotBoxAspectRatioMode','manual','PlotBoxAspectRatio',[1 0.4 1]);
            %set coordinates for Mirror 1 label
            x(1) = mirror1(512,1)-30;
            y(1) = mirror1(512,2)-30;
            %set coordinates for Mirror 2 label
            x(2) = optrans(2,1)+35;
            y(2) = optrans(2,2)-10;
            %set coordinates for Mirror 3 label
            x(3) = optrans(3,1)+35;
            y(3) = optrans(3,2)+10;
            %set coordinates for Hollow Waveguide label
            x(4) = hwgd/2-50;
            y(4) = hwgup + 20;
            %set coordinates for Detector label
            x(5) = optrans(3,1)-r-25;
            y(5) = optrans(3,2)-f-r-10;
            text(x,y,ttext,'FontName','Arial','FontSize',12);
         %          drawnow;
            hold off;
         end;

         photoncount = 0;
         sumdist = 0;
         photondist = zeros(pixels,1);   %distribution of photon distances
```

```
halfbeam = fix(20.0/pixelres);
sourcebeam = 513-halfbeam:513+halfbeam;
beam = 2*halfbeam+1;
mirrordiff = zeros(pixels,1);
mirror3proj = zeros(100,2);
hitmirror3 = 0;
pinctotal = 0;
unilaunch = 1;
if unilaunch
   nphotons = beam;
   startpt = 1:nphotons;
%        startpt = 1:fix(beam/nphotons):nphotons*(fix(beam/nphotons));
end;

missed = zeros(nphotons,1);
mnumber = 0;
for i = 1:nphotons
   %Launch photon
   hitdetect = 0;
   photon = zeros(1024,2);
   if unilaunch
      photonstart = sourcebeam(startpt(i));
   else
      photonstart = sourcebeam(fix(rand(1)*beam)+1);
   end;
   photon(1,:) = [mirror1(photonstart,1) mirror1(1,2)+100];

   %locate where ray hits mirror1, translate to origin and normalize
   nray = zeros(2,2);
   photon(2,:) = mirror1(photonstart,1:2);


   %calculate the reflected ray
   phiw = 0.5*pi;
   if bydiff
      phiv = -(2*mirror1(photonstart,5)-phiw);
      nray(2,:) = [cos(phiv),sin(phiv)];
   else
      phi = 2*(phiw-mirror1(photonstart,5));
      transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
      nray(2,:) = (transmat*nray(1,:)')';
   end;
```

```
            m = nray(2,2)/nray(2,1);
            if  ckangle & abs(m)>m15
               mnumber = mnumber + 1;
               continue;
            end;
            pinc = 2;
            %Check to make sure photon hits hwg hole
            photon(pinc+1,1) = hwgs;
            photon(pinc+1,2) = ((photon(pinc+1,1)-photon(pinc,1))*m)+photon(pinc,2);

            if (photon(pinc+1,2)>=hwgdn)&(photon(pinc+1,2)<=hwgup)
               pinc = pinc+1;
               while (photon(pinc,1)<hwgd);
                  pinc = pinc+1;
                  if m>0
                     photon(pinc,2) = hwgup;
                     photon(pinc,1)  =  ((photon(pinc,2)-photon(pinc-1,2))/m)+photon(pinc-
1,1);
                     m = -m;
                  elseif m<0
                     photon(pinc,2) = hwgdn;
                     photon(pinc,1)  =  ((photon(pinc,2)-photon(pinc-1,2))/m)+photon(pinc-
1,1);
                     m = -m;
                  else
                     photon(pinc,2) = photon(pinc-1,2);
                     photon(pinc,1) = hwgd+1;
                  end;
               end;
               m = -m;
               if m~=0
                  photon(pinc,1) = hwgd;
                  photon(pinc,2)   =   ((photon(pinc,1)-photon(pinc-1,1))*m)+photon(pinc-
1,2);
               else
                  photon(pinc,1) = hwgd;
                  photon(pinc,2) = photon(pinc,2);
               end;
               stopphoton = pinc;
               bpinc   =   round(((((bpx-photon(stopphoton,1))*m)+photon(stopphoton,2))-
botbpy)/beamres);
                  beamprofile(bpinc,runnumber) = beamprofile(bpinc,runnumber) + 1;
```

```
%_____________________________________________________________________
____
% compute pathlength
            phdist = 0;
            for n = 3:pinc-1
               phdist = phdist + pdist(photon(n:n+1,:));
            end;

            %Project to mirror 2
            mirrordiff = ((mirror2(:,1)-photon(pinc,1))*m)+photon(pinc,2);
            if (mirrordiff(1)<=mirror2(1,2))&(mirrordiff(pixels)>=mirror2(pixels,2))
               mirrordiff = (mirror2(:,2)-mirrordiff).^2;
               mirrorpt2 = find(mirrordiff==min(mirrordiff));
               pinc = pinc + 1;
               photon(pinc,:) = mirror2(mirrorpt2,1:2);

               %calculate the reflected ray in mirror2
               nray(1,:) = (photon(pinc-1,:)-mirror2(mirrorpt2,1:2));
               nray(1,:) = nray(1,:)/norm(nray(1,:));
               if m==0;
                  phiw = pi;
               else
                  phiw = atan(m)+pi;
               end;
               if bydiff
                  phiv = 2*mirror2(mirrorpt2,5)-phiw;
                  nray(2,:) = [cos(phiv),sin(phiv)];
               else
                  phi = 2*abs(phiw-mirror2(mirrorpt2,5));
                  transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
                  nray(2,:) = (transmat*nray(1,:)')';
               end;
               m = nray(2,2)/nray(2,1);


               %Project to mirror 3
               mirrordiff = ((mirror3(:,2)-photon(pinc,2))/m)+photon(pinc,1);
         %                  photon(pinc+1,:) = [((60-photon(pinc,2))/m)+photon(pinc,1) 60];
         %                  pinc = pinc + 1;
         if dontskip
               if (mirrordiff(1)<=mirror3(1,1))&(mirrordiff(pixels)>=mirror3(pixels,1))
                  mirrordiff = (mirror3(:,1)-mirrordiff).^2;
                  mirrorpt3 = find(mirrordiff==min(mirrordiff));
                  hitmirror3 = hitmirror3 + 1;
```

```
                pinc = pinc + 1;
                photon(pinc,:) = mirror3(mirrorpt3,1:2);

                %calculate the reflected ray in mirror3
                nray(1,:) = (photon(pinc-1,:)-mirror3(mirrorpt3,1:2));
                nray(1,:) = nray(1,:)/norm(nray(1,:));
                if m==0;
                   phiw = pi;
                else
                   phiw = atan(m)+pi;
                end;
                if bydiff
                   phiv = 2*mirror3(mirrorpt3,5)-phiw;
                   nray(2,:) = [cos(phiv),sin(phiv)];
                else
                   phi = 2*abs(phiw-mirror3(mirrorpt3,5));
                   transmat = [cos(phi) -sin(phi);sin(phi) cos(phi)];
                   nray(2,:) = (transmat*nray(1,:)')';
                end;
        %               reflplot(mirror3(mirrorpt3,5),phiw,phiv,1);
        %               pause(2);
                m = nray(2,2)/nray(2,1);

                %Project to detector
                photon(pinc+1,:)        =        [detcoords(1)        ((detcoords(1)-
photon(pinc,1))*m)+photon(pinc,2)];
                pinc = pinc + 1;
                if
(photon(pinc,2)<=detcoords(2)+detcoords(4))&(photon(pinc,2)>=detcoords(2))
                   sumdist = sumdist + phdist;
                   n = floor((phdist-0)/deltad)+1;
                   photondist(n) = photondist(n) + 1;
                   photoncount = photoncount + 1;
                   hitdetect = 1;
                end;
             end;%if dontskip

          end;    %if hit mirror3
       end;    %if hit mirror2
    end;    %if hit hole

    %Plot photon trace
    if plotphoton & hitdetect
       figure(4);
```

```
            hold on;
            for k = 1:pinc-1
               plot(photon(k:k+1,1),photon(k:k+1,2),'r');
            end;
            hold off;
%           else
%              mnumber = mnumber + 1;
%              missed(mnumber) = startpt(i);
         end;

         pinctotal = pinctotal + pinc-3;
         %plot normal vectors at mirrors
         if plotnormal&plotphoton
            hold on;

      plot([mirror1(photonstart,3)*10+mirror1(photonstart,1)...
            mirror1(photonstart,1)],...
            [mirror1(photonstart,4)*10+mirror1(photonstart,2) ...
            mirror1(photonstart,2)],'-k');

      plot([mirror2(mirrorpt2,3)*10+mirror2(mirrorpt2,1)...
            mirror2(mirrorpt2,1)],...
            [mirror2(mirrorpt2,4)*10+mirror2(mirrorpt2,2) ...
            mirror2(mirrorpt2,2)],'-k');

      plot([mirror3(mirrorpt3,3)*10+mirror3(mirrorpt3,1)...
            mirror3(mirrorpt3,1)],...
            [mirror3(mirrorpt3,4)*10+mirror3(mirrorpt3,2) ...
            mirror3(mirrorpt3,2)],'-k');
                                 hold off;
        end; %if plotnormal
      end;   %for i = photon number
      %       sout = sprintf('Elapsed time:  %6.2f',toc/60);
      %       disp(sout);
      if plotphoton
%              xlim([-200 700]);
%              ylim([-50 250]);
         drawnow;
      end;
      if photoncount>0 avgdist = sumdist/photoncount;
         else avgdist = 0; end;
      sout = sprintf('%4d      %6.2f          %6.2f         %6d         %6.2f         %6.1f
%6.2f',...
         runnumber,m1x,m1y,photoncount,photoncount*100/nphotons,avgdist,toc/60);
```

```
            disp(sout);
            if savedata
               sout = sprintf('%4d      %6.2f          %6.2f        %6d        %6.2f        %6.1f
%6.2f\n',...

runnumber,m1x,m1y,photoncount,photoncount*100/nphotons,avgdist,toc/60);
               fwrite(fid,sout);
            end;
%           end;    %for m1y =
%       end;    %for m1x =
% end;    %for ckangle

if savedata
   fclose(fid);
   eval('save ',fname,'bpdata beamprofile');
end;
```

APPENDIX A-14. HWHW02.M

```
% Generate hollow waveguide

silver = [0.784,0.784,0.784];
HW.radius = 1;
HW.length = 550;
HW.center = [0,25.4,0];
% pixelres is in memory from HWoapm1
% xcoord is the long axis of the cylinder
% version 02 uses MATLAB's cylinder function

[HW.Z,HW.Y,HW.X]                                                                    =
cylinder(ones(fix(HW.radius/pixelres),1)*HW.radius,fix(HW.radius/pixelres));
HW.Y = repmat(HW.Y,[HW.length,1]);
HW.Z = repmat(HW.Z,[HW.length,1]);
Xtmp = HW.X;
for i = 1:HW.length-1
   Xtmp = [Xtmp;HW.X+i];
end;
HW.X = Xtmp;

[Nx,Ny,Nz] = surfnorm(HW.X,HW.Y,HW.Z);
HW.N = repmat(HW.Y,[1,1,3]);
HW.N(:,:,1) = Nx;
HW.N(:,:,2) = Ny;
HW.N(:,:,3) = Nz;

HW.X = HW.X + HW.center(1);
HW.Y = HW.Y + HW.center(2);
HW.Z = HW.Z + HW.center(3);

if plotsurf
   figure(2);
   set(gcf,'Position',[415   524   897   420]);
         surf(HW.X,HW.Y,HW.Z,...
       'EdgeColor','black',...
       'FaceLighting','phong',...
       'FaceColor',silver);
%        axis equal;
         xlabel('X');
         ylabel('Y');
```

```
        zlabel('Z');
    view(0,90);
end;
```

APPENDIX A-15. HWOAPM1D.M

```
% Generate off axis parabolic mirror in similar fashion to SPRAY
% Each paraboloid segment is defined by a location, (x,y,z) of vertex and
%   vectors defining the shape of the paraboloid.  In this case, the y
%   direction is the axis of rotation and its vector is one.  The other two
%   vectors, define the x and z for paraboliod points P(x,1,0) and P(0,1,z).
%   Given the oapm equation, (x^2 + z^2)/4f = y, x and z are the same and
%   equal to 10.0797mm for f=25.4, 90deg oapm.  This implementation is
%   simpler than treating the mirror as a circle.  This is not a problem
%   because the source is circular.  In a future implementation, the
%   necessary parameters will be inputs to an oapm function.
% In this implementation, the box is two dimensional and perpindicular to
%   the axis of rotation.  Therefore the only information needed to define
%   the oapm are the vertex of the paraboloid, the x and z coordinates of
%   the box, the focal length
% The actual segment of the paraboloid that is the oapm is defined by a
%   box that cuts a section of the computed parabolid.
% The paraboloid is contained in a structure P = {xcoord,zcood,Y,N}.
%   M1.xcood contains the coordinates on the x axis.  The length of M1.xcoord is
%   determined by pixelres and the box size in the x direction and is the
%   same size as the number of columns in M1.Y.  Likewise, M1.zcoord are the
%   z coordinates and is the same size as the number of rows in M1.Y.  M1.Y
%   is computed from M1.xcoord, M1.zcoord and f, the focal length.

% version b uses MATLAB's surfnorm to calculate the normals


plotsurf = 0;

copper = [1,0.62,0.40];
f = 25.4;   %focal length of mirror assumed to have a 90deg reflection angle

% pixel resolution is defined by the number of pixels and the 2*length of x
% boxvector.  This size is increased by one to make the coordinates easier
% to calculate in MATLAB using the : operator.
pixels = 1024;
M1.parabcenter = [-50.8,f,0];
M1.boxcenter = [-50.8,f,0];
M1.boxvector = [25.4,25.4,25.4];
pixelres = 2*M1.boxvector(1)/pixels;  %pixel resolution
```

```
xcoord = M1.boxcenter(1)-M1.boxvector(1):pixelres:M1.boxcenter(1)+M1.boxvector(1);
zcoord = (M1.boxcenter(3)-1.boxvector(3):pixelres:M1.boxcenter(3)+M1.boxvector(3))';
xsize = max(size(xcoord));
zsize = max(size(zcoord));

X = repmat(xcoord,zsize,1);
Z = repmat(zcoord,1,xsize);
Y = (X.^2 + Z.^2)./(4*f);

[Nx,Ny,Nz] = surfnorm(X,Y,Z);
N = repmat(Y,[1,1,3]);
N(:,:,1) = Nx;
N(:,:,2) = Ny;
N(:,:,3) = Nz;

M1.X = X + M1.parabcenter(1) - M1.boxcenter(1);
M1.Y = Y + M1.parabcenter(2) - M1.boxcenter(2);
M1.Z = Z + M1.parabcenter(3) - M1.boxcenter(3);
M1.N = N;
M1.center = M1.parabcenter - M1.boxcenter;
M1.minmax = [min(min(M1.X)),max(max(M1.X));...
          min(min(M1.Y)),max(max(M1.Y));...
          min(min(M1.Z)),max(max(M1.Z))];
M1orig = M1;


if plotsurf
   figure(2);
   set(gcf,'Position',[752   524   560   420]);
         surf(M1.X,M1.Y,M1.Z,...
       'EdgeColor','none',...
       'FaceLighting','phong',...
       'FaceColor',copper);
%        axis equal;
         xlabel('X');
         ylabel('Y');
         zlabel('Z');
   view(73,24);
end;
```

APPENDIX A-16. HWOAPM2D.M

```
% Generate off axis parabolic mirror in similar fashion to SPRAY
% Each paraboloid segment is defined by a location, (x,y,z) of vertex and
%   vectors defining the shape of the paraboloid.  In this case, the y
%   direction is the axis of rotation and its vector is one.  The other two
%   vectors, define the x and z for paraboliod points P(x,1,0) and P(0,1,z).
%   Given the oapm equation, (x^2 + z^2)/4f = y, x and z are the same and
%   equal to 10.0797mm for f=25.4, 90deg oapm.  This implementation is
%   simpler than treating the mirror as a circle.  This is not a problem
%   because the source is circular.  In a future implementation, the
%   necessary parameters will be inputs to an oapm function.
% In this implementation, the box is two dimensional and perpindicular to
%   the axis of rotation.  Therefore the only information needed to define
%   the oapm are the vertex of the paraboloid, the x and z coordinates of
%   the box, the focal length
% The actual segment of the paraboloid that is the oapm is defined by a
%   box that cuts a section of the computed parabolid.
% The paraboloid is contained in a structure P = {xcoord,zcood,Y,N}.
%   P.xcood contains the coordinates on the x axis.  The length of P.xcoord is
%   determined by pixelres and the box size in the x direction and is the
%   same size as the number of columns in P.Y.  Likewise, P.zcoord are the
%   z coordinates and is the same size as the number of rows in P.Y.  P.Y
%   is computed from P.xcoord, P.zcoord and f, the focal length.

% version b uses MATLAB's surfnorm to calculate the normals


plotsurf = 0;

copper = [1,0.62,0.40];
f = 25.4;   %focal length of mirror assumed to have a 90deg reflection angle

% pixel resolution is defined by the number of pixels and the 2*length of x
% boxvector.  This size is increased by one to make the coordinates easier
% to calculate in MATLAB using the : operator.
pixels = 1024;
M2.parabcenter = [600.8,f,0];
M2.boxcenter = [50.8,f,0];
M2.boxvector = [25.4,25.4,25.4];
pixelres = 2*M2.boxvector(1)/pixels;  %pixel resolution
```

```
xcoord = M2.boxcenter(1)-M2.boxvector(1):pixelres:M2.boxcenter(1)+M2.boxvector(1);
zcoord = (M2.boxcenter(3)-2.boxvector(3):pixelres:M2.boxcenter(3)+M2.boxvector(3))';
xsize = max(size(xcoord));
zsize = max(size(zcoord));

X = repmat(xcoord,zsize,1);
Z = repmat(zcoord,1,xsize);
Y = (X.^2 + Z.^2)./(4*f);

[Nx,Ny,Nz] = surfnorm(X,Y,Z);
N = repmat(Y,[1,1,3]);
N(:,:,1) = Nx;
N(:,:,2) = Ny;
N(:,:,3) = Nz;

M2.X = X + M2.parabcenter(1) - M2.boxcenter(1);
M2.Y = Y + M2.parabcenter(2) - M2.boxcenter(2);
M2.Z = Z + M2.parabcenter(3) - M2.boxcenter(3);
M2.N = N;
M2.center = M2.parabcenter - M2.boxcenter;
M2.minmax = [min(min(M2.X)),max(max(M2.X));...
             min(min(M2.Y)),max(max(M2.Y));...
             min(min(M2.Z)),max(max(M2.Z))];

if plotsurf
   figure(2);
   set(gcf,'Position',[752   524   560   420]);
          surf(M2.X,M2.Y,M2.Z,...
       'EdgeColor','none',...
       'FaceLighting','phong',...
       'FaceColor',copper);
%        axis equal;
          xlabel('X');
          ylabel('Y');
          zlabel('Z');
   view(110,50);
end;
```

APPENDIX A-17. HWOAPM3D.M

```
% Generate off axis parabolic mirror in similar fashion to SPRAY
% Each paraboloid segment is defined by a location, (x,y,z) of vertex and
%   vectors defining the shape of the paraboloid.  In this case, the y
%   direction is the axis of rotation and its vector is one.  The other two
%   vectors, define the x and z for paraboliod points P(x,1,0) and P(0,1,z).
%   Given the oapm equation, (x^2 + z^2)/4f = y, x and z are the same and
%   equal to 10.0797mm for f=25.4, 90deg oapm.  This implementation is
%   simpler than treating the mirror as a circle.  This is not a problem
%   because the source is circular.  In a future implementation, the
%   necessary parameters will be inputs to an oapm function.
% In this implementation, the box is two dimensional and perpindicular to
%   the axis of rotation.  Therefore the only information needed to define
%   the oapm are the vertex of the paraboloid, the x and z coordinates of
%   the box, the focal length
% The actual segment of the paraboloid that is the oapm is defined by a
%   box that cuts a section of the computed parabolid.
% The paraboloid is contained in a structure P = {xcoord,zcood,Y,N}.
%   P1.xcood contains the coordinates on the x axis.  The length of P1.xcoord is
%   determined by pixelres and the box size in the x direction and is the
%   same size as the number of columns in P1.Y.  Likewise, P1.zcoord are the
%   z coordinates and is the same size as the number of rows in P1.Y.  P1.Y
%   is computed from P1.xcoord, P1.zcoord and f, the focal length.

% version b uses MATLAB's surfnorm to calculate the normals


plotsurf = 0;

copper = [1,0.62,0.40];
f = 25.4;   %focal length of mirror assumed to have a 90deg reflection angle

% pixel resolution is defined by the number of pixels and the 2*length of x
% boxvector.  This size is increased by one to make the coordinates easier
% to calculate in MATLAB using the : operator.
pixels = 1024;
M3.parabcenter = [600.8,f+150,0];
M3.boxcenter = [50.8,-f,0];
M3.boxvector = [25.4,25.4,25.4];
pixelres = 2*M3.boxvector(1)/pixels;  %pixel resolution
```

```
xcoord = M3.boxcenter(1)-M3.boxvector(1):pixelres:M3.boxcenter(1)+M3.boxvector(1);
zcoord = (M3.boxcenter(3)-3.boxvector(3):pixelres:M3.boxcenter(3)+M3.boxvector(3))';
xsize = max(size(xcoord));
zsize = max(size(zcoord));

X = repmat(xcoord,zsize,1);
Z = repmat(zcoord,1,xsize);
Y = -(X.^2 + Z.^2)./(4*f);

[Nx,Ny,Nz] = surfnorm(X,Y,Z);
N = repmat(Y,[1,1,3]);
N(:,:,1) = Nx;
N(:,:,2) = Ny;
N(:,:,3) = Nz;

M3.X = X + M3.parabcenter(1) - M3.boxcenter(1);
M3.Y = Y + M3.parabcenter(2) - M3.boxcenter(2);
M3.Z = Z + M3.parabcenter(3) - M3.boxcenter(3);
M3.N = N;
M3.center = M3.parabcenter - M3.boxcenter;
M3.minmax = [min(min(M3.X)),max(max(M3.X));...
          min(min(M3.Y)),max(max(M3.Y));...
          min(min(M3.Z)),max(max(M3.Z))];


if plotsurf
   figure(2);
   set(gcf,'Position',[752   524   560   420]);
         surf(M3.X,M3.Y,M3.Z,...
       'EdgeColor','none',...
       'FaceLighting','phong',...
       'FaceColor',copper);
%        axis equal;
         xlabel('X');
         ylabel('Y');
         zlabel('Z');
   view(110,50);
end;
```

APPENDIX A-18. HWSOURCE01.M

```
% Generate source

radius = 20;
sourceheight = 5;
startcoord = [-50.8,f+150,0];
% pixelres is in memory from HWoapm1
HW.ycoord = startcoord(2):pixelres:startcoord(2)+sourceheight;
HW.xcoord = 0:pixelres/radius:2*pi;
HW.xcoord = cos(HW.ycoord)*radius + startcoord(3);
HW.zcoord = sin(HW.ycoord)*radius + startcoord(2);
xsize = max(size(HW.xcoord));
ysize = max(size(HW.ycoord));
zsize = max(size(HW.zcoord));

X = repmat(HW.xcoord',1,ysize);
Y = repmat(HW.ycoord,zsize,1);
Z = repmat(HW.zcoord',1,ysize);
Source.X = X;
Source.Y = Y;
Source.Z = Z;

if plotsurf
   figure(2);
   set(gcf,'Position',[415   524   897   420]);
         surf(HW.X,HW.Y,HW.Z,...
       'EdgeColor','black',...
       'FaceLighting','phong',...
       'FaceColor',copper);
%        axis equal;
         xlabel('X');
         ylabel('Y');
         zlabel('Z');
   view(0,90);
end;
```

APPENDIX A-19. TESTHWSIM14.M

```
%testHWsim02 fires a photon from the Source(-50.8,f+150,0) with radius=20
%and reflects the photon through the HW to the exit of the HW
%version 04 changes reflection from 2(N'*I)N - I to I - 2(N'*I)N
% by version 11, mirror surfaces are calculated by surfnorm; projections (t
% for time value) are estimated by solving the quadratic for M2 and simple
% estimate of the y distance for M3; the reflection on the surface point is
% an simplex optimization (MATLAB's fminsearch) of the smallest mu, which
% is the equivalent for the t value of the intersection of a ray and a
% plane defined by its normal.

%Use plotoptics3D to draw 3D mirrors and HW
%then use testHWsim01 to draw a reflected photon off the first mirror only
%photon has three coordinates, [x,y,z], and tracks the path of the photon
clc;

%output controls
plotoptics = 1;
plotphoton = 0;
dispx = 0;
savedata = 0;

%_____________________________________________________________________
____
%header
fname = 'hw3D01';
if savedata fid = fopen([fname,'.txt'],'w'); end;
clc;
disp(' ')
disp('_____________________________________________________________________
_____')
disp(' No.  angle?   M1 x     M1 y    No.Photons  Throughput  Avg Dist     Time   ')
disp('_____________________________________________________________________
_____')
sout = sprintf(' No.  angle?   M1 x     M1 y    No.Photons  Throughput  Avg Dist
Time   \n');
if savedata fwrite(fid,sout); end;



if plotoptics
```

```
   recalc = 1;
   plotoptics3Dc;
end;

%initialize variables
clear global n P D
global n P D
P = zeros(1000,3);
m2coord = zeros(1,3);
m3coord = zeros(1,3);
D = zeros(1000,3);
endphoton = zeros(10000,3);
nphotons = 0;
sourceradius = 20;
sourcelocation = [-50.8,200,0];
hwcoord = zeros(1,3);
maxloopcheck = 2048;
ckangle = 0;
runnumber = 0;
x = 1;
y = 2;
z = 3;
yz = [y,z];
xz = [x,z];
xy = [x,y];
m1y = 0;
m1z = 0;
% M1orig = M1;

% for m1y = -0.2:0.2:0.2
%     for m1z = -0.2:0.2:0.2
      runnumber = runnumber + 1;
                  n = 0;
                  photoncount = 0;
                  phdist = zeros(10000,1);
      m1offset = [0,m1y,m1z];
      M1 = M1orig;
      M1.Y = M1.Y + m1offset(y);
      M1.Z = M1.Z + m1offset(z);
                  tic;

                  for j = 1:nphotons
                        P = zeros(1000,3);
                        D = zeros(1000,3);
```

```
%initialize photon at source
mag = 10;
while mag>1
   xrnd = rand(1)*2 - 1;
   zrnd = rand(1)*2 - 1;
   mag = dot(xrnd,zrnd);
end;
n = 1;
P(n,:) = [xrnd*20+sourcelocation(1),...
            sourcelocation(2),...
            zrnd*20+sourcelocation(3)];
        %       P(n,:) = sourcelocation;
D(n,:) = [0,-1,0];

        %Compute reflection on first mirror to HW
                %compute the difference between mirror magnitudes and photon
magnitude
                mirrordiff = abs(M1.X(1,:) - P(n,1));
                m1coord(x) = find(mirrordiff == min(mirrordiff));
m1coord(x) = m1coord(x);
                mirrordiff = abs(M1.Z(:,1) - P(n,3));
                m1coord(z) = find(mirrordiff == min(mirrordiff));
m1coord(z) = m1coord(z);

                %update nphoton
                P(n,:)                                                          =
[M1.X(m1coord(z),m1coord(x)),P(n,2),M1.Z(m1coord(z),m1coord(x))];
n = n + 1;
                P(n,:)                                                          =
[M1.X(m1coord(z),m1coord(x)),M1.Y(m1coord(z),m1coord(x)),M1.Z(m1coord(z),m1co
ord(x))];

                %compute the normal and incoming vectors [x,y,z]
                mirrornorm = shiftdim(M1.N(m1coord(z),m1coord(x),:))';
                D(n,:) = D(n-1,:) - 2*dot(D(n-1,:),mirrornorm)*mirrornorm;

        %compute location of first intersection with HW

                %compute projection of outgoing ray to x=0
%(needed to compute pathlength)
                nextx = 0;
                t = (nextx-P(n,x))/D(n,x);
n = n + 1;
                P(n,:) = P(n-1,:) + t*D(n-1,:);
```

```
D(n,:) = D(n-1,:);

%display current x position
if dispx
   clc;
   s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
   disp(s);
end;

%Check for inside HW diameter
%Compute length of y,z vector from center of HW to photon locate at x=0
loopcheck = 0;
hit = 0;
tmp = norm(P(n,yz)-HW.center(yz));
if tmp < HW.radius hit = 1; end;
while hit & loopcheck<maxloopcheck
   loopcheck = loopcheck + 1;
   C = sum((P(n,yz)-HW.center(yz)).^2) - HW.radius^2;
   B = 2*dot(P(n,yz)-HW.center(yz),D(n,yz));
   A = D(n,yz)*D(n,yz)';
   discrim = B^2 - (4*A*C);
   if discrim < 0 hit = 0;
   else
      sqdiscrim = sqrt(discrim);
      t = [(-B + sqdiscrim)/(2*A),(-B - sqdiscrim)/(2*A)];
      tmp = repmat(P(n,:),2,1) + t'*D(n,:);
      next_t = find(tmp(:,x)>=P(n,x) + 1e-6);
      if isempty(next_t) hit = 0;
      else
         if tmp(next_t,x)>=HW.length
            t = (HW.length - P(n,x))/D(n,x);
            hit = 0;
            n = n + 1;
            P(n,:) = P(n-1,:) + t*D(n-1,:);
            D(n,:) = D(n-1,:);
            %display current x position
            if dispx
               clc;
               s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
               disp(s);
            end;
         else
            %determine cylinder pixelated intersection point
            t = t(next_t);
```

```
                    n = n + 1;
                    P(n,:) = P(n-1,:) + t*D(n-1,:);
                    mirrordiff = (HW.X(:,1) - P(n,1)).^2;
                    tmp = find(mirrordiff == min(mirrordiff));
                    hwcoord(x) = tmp(1);
                    mirrordiff = sum([HW.Y(hwcoord(x),:)-P(n,y);HW.Z(hwcoord(x),:)-
P(n,z)].^2);
                    tmp = find(mirrordiff == min(mirrordiff));
                    hwcoord(y) = tmp(1);
                    P(n,:)                                                    =
[HW.X(hwcoord(x),hwcoord(y)),HW.Y(hwcoord(x),hwcoord(y)),HW.Z(hwcoord(x),hw
coord(y))];
                    %display current x position
                    if dispx
                       clc;
                       s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
                       disp(s);
                    end;
                    mirrornorm = shiftdim(HW.N(hwcoord(x),hwcoord(y),:))';
                    %compute reflection at cylinder intersection point
                                                        D(n,:)  =  D(n-1,:)  -  2*dot(D(n-
1,:),mirrornorm)*mirrornorm;
                 end;
              end;
           end;
           if ((P(n,:)*P(n-1,:)')<1e-6) return; end;
        end;    %while hit

        if loopcheck>maxloopcheck error('Stuck in while hit loop'); end;

        %compute pathlength
        HWexit = n;
        for pinc = 3:HWexit-1
           phdist(n) = phdist(n) + pdist(P(pinc:pinc+1,:));
        end;


        %Project ray to mirror 2
        %translate to original function
        raystart = P(n,:) - M2.center;
        C = sum(raystart(xz).^2) - (4*f*raystart(y));
        B = 2*(dot(raystart(xz),D(n,xz)) - (4*f*D(n,y)));
        A = D(n,xz)*D(n,xz)';
        discrim = B^2 - (4*A*C);
```

```
if discrim < 0
   endphoton(j,:) = P(n,:);
   continue;
else
   sqdiscrim = sqrt(discrim);
   t = [(-B + sqdiscrim)/(2*A),(-B - sqdiscrim)/(2*A)];
   tmp = repmat(raystart,2,1) + t'*D(n,:) + repmat(M2.center,2,1);
   next_t = find(tmp(:,x)>=P(n,x) + 1e-6);
   if isempty(next_t)
      endphoton(j,:) = P(n,:);
      continue;
   else
      %determine mirror pixelated intersection point
      t2 = t(find(t.^2==min(t.^2)));
      t2 = fminsearch(@testmu4,t2,[],M2);
      n = n + 1;
      hit = [(P(n,:)'>M2.minmax(:,1)),(P(n,:)'<M2.minmax(:,2))];
      if ~all(hit(:))
         endphoton(j,:) = P(n,:);
         continue;
      else
         m2coord = findcoords(P(n,:),M2.X(1,:),M2.Z(:,1));
         P(n,:)= [M2.X(m2coord(z),m2coord(x)),...
                  M2.Y(m2coord(z),m2coord(x)),...
                  M2.Z(m2coord(z),m2coord(x))];
         %display current x position
         if dispx
            clc;
            s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
            disp(s);
         end;
         N = shiftdim(M2.N(m2coord(z),m2coord(x),:))';
         %compute reflection at mirror intersection point

                                          D(n-1,:) = P(n,:) - P(n-1,:);
                                          D(n-1,:) = D(n-1,:)/norm(D(n-1,:));
                                          D(n,:) = D(n-1,:) - 2*dot(D(n-1,:),N)*N;
      end;
   end;
end;

%Project ray to mirror 3
%translate to original function
                  t = (M3.parabcenter(y) + M2.parabcenter(y) - 2*P(n,y))/D(n,y);
```

```
                tmp = P(n,:) + t*D(n,:);
hit = [(tmp'>M3.minmax(:,1)),(tmp'<M3.minmax(:,2))];
if ~all(hit(:))
   endphoton(j,:) = P(n,:);
   continue;
else
        %       t3 = fminsearch(@testmu4,t3,optimset('Display','iter'),M3);
   t3 = t;
   t3 = fminsearch(@testmu4,t3,[],M3);
   n = n + 1;
   m3coord = findcoords(P(n,:),M3.X(1,:),M3.Z(:,1));
   P(n,:)= [M3.X(m3coord(z),m3coord(x)),...
         M3.Y(m3coord(z),m3coord(x)),...
         M3.Z(m3coord(z),m3coord(x))];
   %display current x position
   if dispx
      clc;
      s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
      disp(s);
   end;
   N = shiftdim(M3.N(m3coord(z),m3coord(x),:))';
   %compute reflection at mirror intersection point
                        D(n-1,:) = P(n,:) - P(n-1,:);
                        D(n-1,:) = D(n-1,:)/norm(D(n-1,:));
                        D(n,:) = D(n-1,:) - 2*dot(D(n-1,:),N)*N;
end;

% compute intersection with detector
t = (Det.center(x)-P(n,x))/D(n,x);
tmp = P(n,:) + t*D(n,:);
hit = [(tmp>=Det.minmax(1,:));(tmp<=Det.minmax(2,:))];
if all(hit(:))
   n = n + 1;
   P(n,:) = tmp;
   photoncount = photoncount + 1;
   if dispx
      clc;
      s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
      disp(s);
   end;
end;

endphoton(j,:) = P(n,:);
```

```
                end;

                % tout = toc/60;
                %plot photon path
                if plotphoton
                        figure(2);
                        hold on;
        for i = 2:n
                                plot3(P(i-1:i,1),P(i-1:i,2),P(i-1:i,3),'Color','r','LineWidth',1);
                        end;
                        hold off;
                end;


                if photoncount>0 avgdist = mean(phdist(find(phdist)));
        else avgdist = 0; end;
                sout = sprintf('%4d     %1d     %6.2f    %6.2f     %6d         %6.2f     %6.1f
%6.2f',...

runnumber,ckangle,m1y,m1z,photoncount,photoncount*100/nphotons,avgdist,toc/60);
                disp(sout);
                if savedata
        sout = sprintf('%4d       %1d       %6.2f     %6.2f       %6d           %6.2f       %6.1f
%6.2f\n',...

runnumber,ckangle,m1y,m1z,photoncount,photoncount*100/nphotons,avgdist,toc/60);
        fwrite(fid,sout);
                end;

%       end;    %for m1z
% end;    %for m1y

%_____________________________________________________________________________
____
% tidy up
if savedata
   fclose(fid);
end;
```

APPENDIX A-20. TESTHWSIM15WC.M

```
%testHWsim02 fires a photon from the Source(-50.8,f+150,0) with radius=20
%and reflects the photon through the HW to the exit of the HW
%version 04 changes reflection from 2(N'*I)N - I to I - 2(N'*I)N
% by version 11, mirror surfaces are calculated by surfnorm; projections (t
% for time value) are estimated by solving the quadratic for M2 and simple
% estimate of the y distance for M3; the reflection on the surface point is
% an simplex optimization (MATLAB's fminsearch) of the smallest mu, which
% is the equivalent for the t value of the intersection of a ray and a
% plane defined by its normal.

%Use plotoptics3D to draw 3D mirrors and HW
%then use testHWsim01 to draw a reflected photon off the first mirror only
%photon has three coordinates, [x,y,z], and tracks the path of the photon
clc;

%_____
____
% output controls
plotoptics = 0;
plotphoton = 0;
dispx = 0;
savedata = 0;
calcoptics = 0;

%_____
____
% header
fname = 'hw3Dwc';
if savedata fid = fopen([fname,'.txt'],'w'); end;
clc;
disp(' ')
disp('_____
_____')
disp(' No.  angle?   a    ap    No.Photons  Throughput  Avg Dist     Time   ')
disp('_____
_____')
sout = sprintf(' No.  angle?   a    ap    No.Photons  Throughput  Avg Dist     Time
\n');
if savedata fwrite(fid,sout); end;
```

```
%_____
____
clear global n P D
global n P D

% initialize optics
if plotoptics
   recalc = 0;
   plotoptics3Dd;
end;

if ~plotoptics & calcoptics
        HWoapm1d;
        HWhw02;
   wc = makeplc(2,0.4,pixelres,[HW.length+10,f,0]);
   HWdetwc;
   clear X Y Z xcoord zcoord Nx Ny Nz Xtmp
end;

%_____
____
% initialize variables
P = zeros(1000,3);
m1coord = zeros(1,3);
wccoord = zeros(1,3);
D = zeros(1000,3);
endphoton = zeros(10000,3);
nphotons = 10000;
sourceradius = 20;
sourcelocation = [-50.8,200,0];
hwcoord = zeros(1,3);
maxloopcheck = 2048;
ckangle = 1;
tan15 = tan(15/180);
runnumber = 0;
hitdet = 0;
x = 1;
y = 2;
z = 3;
yz = [y,z];
xz = [x,z];
xy = [x,y];
```

```
m1y = 0;
m1z = 0;
% M1orig = M1;

%_____________________________________________________________________
____
% Begin configuration iteration looping thingy
% for ckangle = 0:1
a = 2;
ap = 0.4;
%  for a = 2:0.4:7.6
%     for ap = 0.1:0.1:1
   wcorig = makeplc(a,ap,pixelres,[HW.length,f,0]);
   wcoffset = 10;
      runnumber = runnumber + 1;
                  n = 0;
                  photoncount = 0;
                  phdist = zeros(10000,1);
      wc = wcorig;
      wc.X = wc.X + wcoffset;
      HWdetwc;
                  tic;

%_____________________________________________________________________
____
% Launch photons
naccept = 0;
lphoton = 0;
                  for j = 1:nphotons
                        P = zeros(1000,3);
                        D = zeros(1000,3);
         %initialize photon at source
         mag = 10;
         while mag>1
            xrnd = rand(1)*2 - 1;
            zrnd = rand(1)*2 - 1;
            mag = dot(xrnd,zrnd);
         end;
         n = 1;
         P(n,:) = [xrnd*20+sourcelocation(1),...
                     sourcelocation(2),...
                     zrnd*20+sourcelocation(3)];
                  %        P(n,:) = sourcelocation;
         D(n,:) = [0,-1,0];
```

```
%_____
____
% Compute reflection on first mirror to HW
                        %compute the difference between mirror magnitudes and photon
magnitude
                        mirrordiff = abs(M1.X(1,:) - P(n,1));
                        m1coord(x) = find(mirrordiff == min(mirrordiff));
        m1coord(x) = m1coord(x);
                        mirrordiff = abs(M1.Z(:,1) - P(n,3));
                        m1coord(z) = find(mirrordiff == min(mirrordiff));
        m1coord(z) = m1coord(z);

                        %update nphoton
                        P(n,:)                                                  =
[M1.X(m1coord(z),m1coord(x)),P(n,2),M1.Z(m1coord(z),m1coord(x))];
        n = n + 1;
                        P(n,:)                                                  =
[M1.X(m1coord(z),m1coord(x)),M1.Y(m1coord(z),m1coord(x)),M1.Z(m1coord(z),m1co
ord(x))];

                        %compute the normal and incoming vectors [x,y,z]
                        mirrornorm = shiftdim(M1.N(m1coord(z),m1coord(x),:))';
                        D(n,:) = D(n-1,:) - 2*dot(D(n-1,:),mirrornorm)*mirrornorm;

%_____
____
% check acceptance angle
%           accept = 0;
        accept = ((sqrt(dot(D(n,yz),D(n,yz)))/D(n,x)) <= tan15);
        if ckangle & ~accept
           naccept = naccept + 1;
           continue;
        end;
        lphoton = lphoton + 1;

%_____
____
% compute location of first intersection with HW
                        %compute projection of outgoing ray to x=0
        %(needed to compute pathlength)
                        nextx = HW.center(x);
                        t = (nextx-P(n,x))/D(n,x);
        n = n + 1;
```

```
                    P(n,:) = P(n-1,:) + t*D(n-1,:);
        D(n,:) = D(n-1,:);

        %display current x position
        if dispx
           clc;
           s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
           disp(s);
        end;

%_____
____
% Check for inside HW diameter
        %Compute length of y,z vector from center of HW to photon locate at x=0
        loopcheck = 0;
        hit = 0;
        tmp = norm(P(n,yz)-HW.center(yz));
        if tmp < HW.radius hit = 1; else continue; end;

%_____
____
% Trace through hollow waveguide
        while hit & loopcheck<maxloopcheck
           loopcheck = loopcheck + 1;
           C = sum((P(n,yz)-HW.center(yz)).^2) - HW.radius^2;
           B = 2*dot(P(n,yz)-HW.center(yz),D(n,yz));
           A = D(n,yz)*D(n,yz)';
           discrim = B^2 - (4*A*C);
           if discrim < 0 hit = 0;
           else
              sqdiscrim = sqrt(discrim);
              t = [(-B + sqdiscrim)/(2*A),(-B - sqdiscrim)/(2*A)];
              tmp = repmat(P(n,:),2,1) + t'*D(n,:);
              next_t = find(tmp(:,x)>=P(n,x) + 1e-6);
              if isempty(next_t) hit = 0;
              else
                 if tmp(next_t,x)>=HW.length
                    t = (HW.length - P(n,x))/D(n,x);
                    hit = 0;
                    n = n + 1;
                    P(n,:) = P(n-1,:) + t*D(n-1,:);
                    D(n,:) = D(n-1,:);
                    %display current x position
                    if dispx
```

```
                    clc;
                    s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
                    disp(s);
                end;
            else
                %determine cylinder pixelated intersection point
                t = t(next_t);
                n = n + 1;
                P(n,:) = P(n-1,:) + t*D(n-1,:);
                mirrordiff = (HW.X(:,1) - P(n,1)).^2;
                tmp = find(mirrordiff == min(mirrordiff));
                hwcoord(x) = tmp(1);
                mirrordiff   =   sum([HW.Y(hwcoord(x),:)-P(n,y);HW.Z(hwcoord(x),:)-
P(n,z)].^2);
                tmp = find(mirrordiff == min(mirrordiff));
                hwcoord(y) = tmp(1);
                P(n,:)                                                              =
[HW.X(hwcoord(x),hwcoord(y)),HW.Y(hwcoord(x),hwcoord(y)),HW.Z(hwcoord(x),hw
coord(y))];
                %display current x position
                if dispx
                    clc;
                    s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
                    disp(s);
                end;
                mirrornorm = shiftdim(HW.N(hwcoord(x),hwcoord(y),:))';
                %compute reflection at cylinder intersection point
                                                    D(n,:)   =   D(n-1,:)   -   2*dot(D(n-
1,:),mirrornorm)*mirrornorm;
                end;
            end;
        end;
        if ((P(n,:)*P(n-1,:)')<1e-6) return; end;
    end;    %while hit

    if loopcheck>maxloopcheck error('Stuck in while hit loop'); end;

%_____
____
% compute pathlength
    HWexit = n;
    for pinc = 3:HWexit-1
        phdist(j) = phdist(j) + pdist(P(pinc:pinc+1,:));
    end;
```

```
%_____
____
% PLC (Parabolic Light Concentrator or Winston Cone)
        %Project ray to PLC
        %compute point at entrance of PLC
                        t = (wc.X(x,1)-P(n,x))/D(n,x);
        n = n + 1;
                        P(n,:) = P(n-1,:) + t*D(n-1,:);
        D(n,:) = D(n-1,:);

        %check if ray hits PLC interior
        inwc = 0;
        tmp = norm(P(n,yz)-wc.center(yz));
        if tmp <= wc.curve(1,2) inwc = 1; else continue; end;

        %trace ray through PLC interior
        loopcheck = 0;
        while inwc & loopcheck<maxloopcheck
           loopcheck = loopcheck + 1;
           %check for ray reflecting back out of PLC
           dirP = P(n,x) - P(n-1,x);

           %compute point at exit of PLC
                                t = (wc.X(wc.pixels,x)-P(n,x))/D(n,x);
                                tmp = P(n,:) + t*D(n,:);

           %check if ray goes straight through PLC
           yzvector = norm(tmp(yz)-wc.center(yz));
           if (yzvector < wc.curve(wc.pixels,2))|(dirP<=0)
              inwc = 0;
           else
              %compute t and r (yz vector length) for all x of PLC
              t = (wc.X(:,1) - P(n,x))./D(n,x);
              yztmp        =        sum(((repmat(P(n,yz)-wc.center(yz),wc.pixels,1)        +
t*D(n,yz)).^2),2);
              tmp = (wc.curve(:,2) - yztmp).^2;
              next_t = find(tmp==min(tmp));
              next_t = next_t(1);
              if ((t(next_t)==0) & (next_t<wc.pixels)) next_t = next_t + 1; end;
              if ((next_t==1)|(next_t>=wc.pixels)) & ...
                    yztmp(next_t)<wc.curve(next_t,2)
                 inwc = 0;
```

```
            else
               twc = t(next_t);
               n = n + 1;
               P(n,:) = P(n-1,:) + twc*D(n-1,:);
               wccoord = findcoordswc(P(n,:),wc);
               P(n,:)= [wc.X(wccoord(x),wccoord(y)),...
                        wc.Y(wccoord(x),wccoord(y)),...
                        wc.Z(wccoord(x),wccoord(y))];
               %display current x position
               if dispx
                  clc;
                  s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
                  disp(s);
               end;
               N = shiftdim(wc.N(wccoord(x),wccoord(y),:))';
               %compute reflection at mirror intersection point
                                                D(n-1,:) = P(n,:) - P(n-1,:);
               ND = norm(D(n-1,:));
%                 if ND == 0 error('D normal = 0'); end;
                                                D(n-1,:) = D(n-1,:)/ND;
                                                D(n,:) = D(n-1,:) - 2*dot(D(n-1,:),N)*N;
            end;    %if on wc surface
         end;    %if yzvector & dirP
      end;    %while inwc

      if loopcheck>maxloopcheck error('Stuck in while hit loop'); end;



%_____________________________________________________________________
____
% Detector projection
      % compute intersection with detector
      hitdet = 0;
      t = (Det.center(x)-P(n,x))/D(n,x);
      tmp = P(n,:) + t*D(n,:);
      hit = [(tmp>=Det.minmax(1,:));(tmp<=Det.minmax(2,:))];
      if all(hit(:))
         hitdet = 1;
         n = n + 1;
         P(n,:) = tmp;
         photoncount = photoncount + 1;
         if dispx
            clc;
```

```
            s = sprintf('Current position:  %5.1f  %5.1f  %5.1f',P(n,:));
            disp(s);
        end;
    end;
%_____
____
% plot photon path
                    if plotphoton & hitdet
                        figure(2);
                        hold on;
        for i = 2:n
                                plot3(P(i-1:i,1),P(i-1:i,2),P(i-
1:i,3),'Color','r','LineWidth',1);
                        end;
                        hold off;
                    end;

    endphoton(j,:) = P(n,:);

            end;    %end for i = 1:nphotons

%_____
____
% summary calculations
            % tout = toc/60;


            if photoncount>0 avgdist = mean(phdist(find(phdist)));
    else avgdist = 0; end;
            sout = sprintf('%4d    %1d    %6.2f    %6.2f    %6d        %6.2f    %6.1f
%6.2f',...
    runnumber,ckangle,a,ap,photoncount,photoncount*100/nphotons,avgdist,toc/60);
            disp(sout);
            if savedata
    sout = sprintf('%4d      %1d      %6.2f     %6.2f      %6d          %6.2f      %6.1f
%6.2f\n',...
    runnumber,ckangle,a,ap,photoncount,photoncount*100/nphotons,avgdist,toc/60);
    fwrite(fid,sout);
            end;

  end;    %for wcoffset
end;    %for a
% end;    %for ckangle
```

```
%_____
____
% tidy up
if savedata
   fclose(fid);
end;
```